Contactless sub-millimeter displacement measurements

Guus Sliepen[a,b], Aswin P.L. Jägers[a,b], Felix C.M. Bettonvil[a,b] and Robert H. Hammerschlag[a]

[a]Astronomical Institute, Utrecht University, Princetonplein 5, 3584 CC Utrecht, the Netherlands
[b]Technology Foundation STW, Utrecht, the Netherlands

ABSTRACT

Weather effects on foldable domes, as used at the DOT and GREGOR, are investigated, in particular the correlation between the wind field and the stresses caused to both metal framework and tent clothing. Camera systems measure contactless the displacement of several dome points. The stresses follow from the measured deformation pattern. The cameras placed near the dome floor do not disturb telescope operations. In the set-ups of DOT and GREGOR, these cameras are up to 8 meters away from the measured points and must be able to detect displacements of less than 0.1 mm. The cameras have a FireWire (IEEE1394) interface to eliminate the need for frame grabbers. Each camera captures 15 images of 640×480 pixels per second. All data is processed on-site in real-time. In order to get the best estimate for the displacement within the constraints of available processing power, all image processing is done in Fourier-space, with all convolution operations being pre-computed once. A sub-pixel estimate of the peak of the correlation function is made. This enables to process the images of four cameras using only one commodity PC with a dual-core processor, and achieve an effective sensitivity of up to 0.01 mm. The deformation measurements are well correlated to the simultaneous wind measurements. The results are of high interest to upscaling the dome design (ELTs and solar telescopes).

Keywords: Foldable tent domes, Contactless displacement, Data reduction

1. INTRODUCTION

As part of a research project to investigate the behavior of completely open foldable domes in strong winds, sensor units have been developed to measure the deformation of the existing domes at the Dutch Open Telescope[4] (DOT) on La Palma and the GREGOR telescope[5] on Tenerife. Although stress on the construction could be measured in a more direct way, for example with strain gauges, such sensors are difficult to install after construction. In addition, these sensors give only very low local differential information about the deformations. The measured values are strongly dependent on geometrical details of the construction. Consequently, a large number of these sensors would be required for a clear insight into load and stress of the dome as a whole. Most importantly, since any modifications to the dome itself were to be avoided, it was decided to measure the deformation with contactless sensors.

These foldable domes consist of cloth made of tensile material spanned over steel bows.[1] In strong winds, the domes are completely closed to protect the telescopes inside them. Because of the stiffness of the construction, the displacement of any point of the dome, even in very strong winds, will not be more than several centimeters. In order to also see displacement at lower wind speed, we want to achieve a sensitivity of 0.1 mm or better. The domes are typically many magnitudes larger (7 m in diameter or more). It is therefore infeasible to use, for example, a single camera to measure the deformation of the entire dome. Instead, we will measure the displacement of several points on the cloth and bows of the dome, and reconstruct the entire deformation from these individual points.

For these individual points, the curvature of the dome can be neglected. We can also neglect rotation and shearing and we will only measure the translation. The translation of a point on the dome can be in any direction; either in its own plane or perpendicular to it (see figure 1).

Since laser or ultrasonic rangers project light or sound onto a surface, they cannot be used to measure in-plane displacement. It was therefore decided to use cameras to track the displacement of individual points on the dome. Although a single camera can only measure displacement in two dimensions, with two cameras looking at the same point, stereo decomposition can be used to determine displacement in three dimensions. Compact flat units have been developed in

Send correspondence to G. Sliepen <G.Sliepen@uu.nl>, phone: +31 (0)30 253 1617, web: http://dot.astro.uu.nl/

Advanced Optical and Mechanical Technologies in Telescopes and Instrumentation, edited by Eli Atad-Ettedgui, Dietrich Lemke, Proc. of SPIE Vol. 7018, 70181C, (2008) · 0277-786X/08/$18 · doi: 10.1117/12.788320

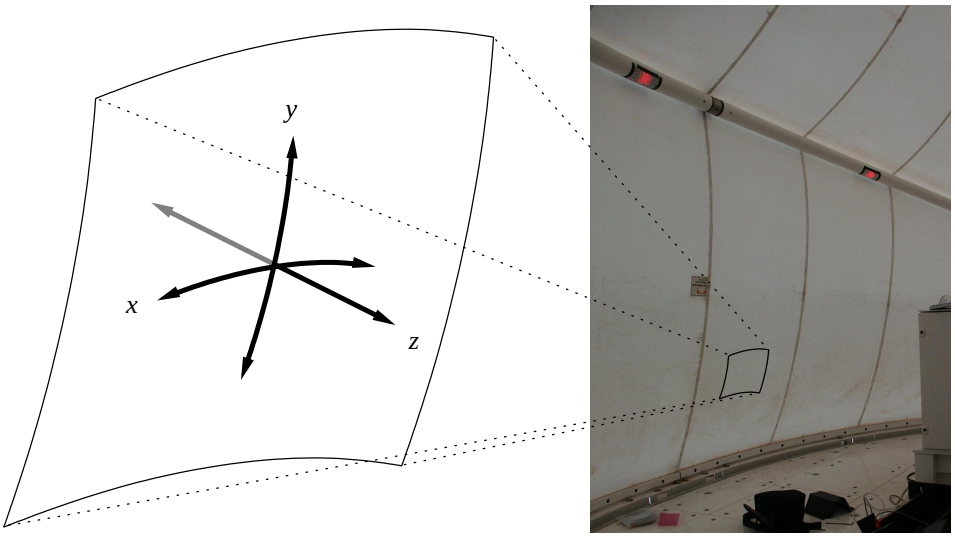

Figure 1. A section of tent cloth can move due to wind exerting force on the dome. This can either be in the plane of the cloth itself (*x* and *y* directions) or out of the plane (*z* direction). The cloth is pre-tensioned and has a saddle shape, but at length scales of several centimeters the surface can be considered flat.

which the stereo base is realised with two flat mirrors and two lenses in front of the cameras, the so-called 3-Dimensional Dome Displacement Detector (3DD3) units.[2]

In the next section we will explain our motivations for our choice of camera. In section 3 we will describe the algorithms used to pre-process the camera images and to calculate the displacement from them. In section 4 and 5 we will describe our implementation and some of the results from actual measurements. We present our conclusion in section 6.

2. CAMERAS

In order to recover the deformation of the entire dome from the displacement of the individual points, a substantial number of points needs to be measured. This means that affordable cameras had to be chosen that could be read out in an easy way. For this reason we have chosen the Unibrain Fire-i™ board camera. The specifications of this type of camera are given in table 1.

When optics are chosen such that 1 camera pixel equals 0.1 mm on the dome, the number of pixels allows displacements of up to 6.4 cm horizontally and 4.8 cm vertically to be measured. The maximum framerate of 30 fps allows us to measure oscillations of up to 15 Hz. This camera has two FireWire (IEEE1394) ports that allow us to daisy chain several cameras without needing hubs or repeaters. Since FireWire is a common standard, and many computer motherboards are equipped with them, no extra hardware, such as a framegrabber, is needed to read out these cameras. Tests have shown that due to bandwidth limitations, on a single FireWire 400 bus only approximately 95 frames per second can be transmitted with cameras of this type. Furthermore, the FireWire controllers in most computers can only handle up to four isochronous channels. Since each camera uses one isochronous channel, up to 4 cameras can be connected to a single bus. We currently connect 4 cameras to one computer, each running at 15 fps, but we can also have two of the four cameras running at 30 fps.

The Fire-i™ cameras use the IIDC protocol which is used by many industrial FireWire cameras. The standardization in this protocol goes much further than for example Camera Link®, and allows one to exchange cameras for those of a completely different brand without having to change the acquisition software.

Measurement units have been made that contain two of these cameras so that displacement can be detected in three dimensions.[2]

Horizontal resolution:	640 pixels
Vertical resolution:	480 pixels
Pixel size:	5.6×5.6 µm
Color depth:	8-bit grayscale
Maximum framerate:	30 fps
Interface:	2 6-pin FireWire 400 ports
Protocol:	IIDC v1.04

Table 1. Specifications of the Unibrain Fire-i™ board camera.

3. DISPLACEMENT DETECTION

The displacement of a point on the dome is related to wind pressure on the dome. We are interested in rapid deformation caused by strong gusts, but also in slow, possibly constant deformation caused by enduring wind. We therefore want to measure the absolute displacement of points compared to their reference positions taken at a time when there was no wind pressure.

At times of strong wind, which occur especially in the winter season, no persons are likely to be in the dome. The sensor units must therefore be able to operate autonomously for months. Lighting conditions change during the day though, and dust can accumulate on optics over time. To reduce complexity and costs, the units have no movable lenses or filters. It is therefore not possible to do flat- or darkfield measurements automatically. Instead, a time-independent image filter has to be used to reduce the effects of gradients caused by lighting conditions and dust accumulation on the optics.

Since the data output from a single camera running at 15 fps is already in excess of 370 GB per day, it is completely unfeasible to store the camera images for later processing. Image processing has to be done in real time. It is therefore critical that this is done as efficiently as possible.

3.1. Acquiring reference images

Before displacement can be measured, a reference image for each camera has to be acquired at a time when there is no deformation (i.e. very low to zero wind speed). To reduce noise, a series of images can be captured and averaged, although displacements of more than a pixel during the series can reduce the sharpness of the resulting average. Information from on-site anemometers can be used to determine when there are periods of low wind speeds. When averaging, simple correlation tracking can be used to determine the amplitude of displacement during capturing of the image series. A series can be discarded if the amplitude exceeds a preset value. With these simple heuristics it is possible to completely automate the generation of reference images.

3.2. Calculating the correlation function

A fast way to calculate the correlation function C of a reference image I_{ref} and new image I_{new} is to Fourier transform both images, calculate the piecewise product of one image with the complex conjugate of the other, and transform the result back:

$$C = I_{\text{ref}} \otimes I_{\text{new}} = \mathscr{F}^{-1}\left[\mathscr{F} I_{\text{ref}} \cdot (\mathscr{F} I_{\text{new}})^{\dagger}\right] \tag{1}$$

3.3. Gradient removal and sharpening

For correlation tracking we are more interested in sharp features than in features varying slowly in intensity. We also want to eliminate gradients caused by lighting, since these typically do not move together with the cloth. We can eliminate the low frequency components of the image by subtracting a blurred version of the image from itself. Such an operation can be written as a convolution of the original image I_{ref} and the filter kernel F:

$$I_{\text{ref}} \odot F = \mathscr{F}^{-1}\left[\mathscr{F} F \cdot \mathscr{F} I_{\text{ref}}\right] \tag{2}$$

Since calculating the correlation function is already done using Fourier transforms, we can use the fact that multiplication is commutative and associative. A filter that needs to be applied to both images that are to be correlated, can instead also be applied twice to only one of the images:

$$C_{\text{filtered}} = \mathscr{F}^{-1}\left[(\mathscr{F}F \cdot \mathscr{F}I_{\text{ref}}) \cdot (\mathscr{F}F \cdot \mathscr{F}I_{\text{new}})^{\dagger}\right] = \mathscr{F}^{-1}\left[(\mathscr{F}F)^2 \cdot \mathscr{F}I_{\text{ref}} \cdot (\mathscr{F}I_{\text{new}})^{\dagger}\right] \tag{3}$$

3.4. Deconvolution of the correlation function

The correlation function will in general look like the autocorrelation function $A = I_{\text{ref}} \otimes I_{\text{ref}}$ of the reference image, but translated over the same distance as the image being correlated to the reference image. The form of the autocorrelation function depends on the texture of the cloth or bow at the point on the dome the camera is looking at. The autocorrelation will in general not be a delta peak but spread out. The symmetry of the form of the peak also depends on the symmetries of the cloth texture. Theoretically, it would be possible to deconvolve the correlation function with the autocorrelation function, so the result would be a delta peak at exactly the spot corresponding to the translation of the cloth. This is however not possible in the presence of noise and other imperfections. However, when a good estimate can be made of the noise, Wiener filtering can be used to construct an optimal filter Φ that can be applied prior to the deconvolution to counteract the effects of noise.[3] The deconvolved correlation function is given by the following equation:

$$C_{\text{Wiener}} = \mathscr{F}^{-1}\left[\frac{\mathscr{F}C \cdot \Phi}{\mathscr{F}A}\right] \tag{4}$$

The result will bring the correlation function closer to a perfect delta peak. An estimate of the noise in the image could be obtained automatically by calculating the standard deviation of each frequency component when averaging the reference image. However, the amount of noise is dependent on the gain, exposure time and temperature of the CCD, and may vary considerably over time. Since Wiener filtering can be implemented as a multiplication and division in Fourier space, it is possible to apply it to the reference image once instead of applying to the correlation function as a whole.

3.5. Apodization

To reduce the effect of discontinuities at the edges of the images, apodization is applied to one of the images to be correlated. It does not matter for correlation if we apply apodization to the reference image or the new image, therefore we apply it to the reference image. The apodization filter W is applied in the spatial domain:

$$C_{\text{apodized}} = \mathscr{F}^{-1}\left[\mathscr{F}(W \cdot I_{\text{ref}}) \cdot \mathscr{F}(I_{\text{new}})^{\dagger}\right] \tag{5}$$

3.6. Precomputation

We can combine all three techniques described above and combine the sharpening filter F, the Wiener filter Φ, the deconvolution with the autocorrelation function A and the apodization window W into a single pre-computed filter P:

$$P = \mathscr{F}\left(W \cdot \mathscr{F}^{-1}\left[\frac{(\mathscr{F}F)^2 \cdot \mathscr{F}I_{\text{ref}} \cdot \Phi}{\mathscr{F}A}\right]\right) \tag{6}$$

From then on, new camera images can be acquired, and calculating the correlation function of the filtered images is reduced to two Fourier transforms and a piecewise, complex conjugated multiplication:

$$C = \mathscr{F}^{-1}\left[P \cdot (\mathscr{F}I_{\text{new}})^{\dagger}\right] \tag{7}$$

3.7. Sub-pixel estimation

For each pixel the correlation function effectively gives the probability that the new image is translated over a distance equal to the position of that pixel. The pixel with the highest correlation gives us the most probably translation vector, with pixel resolution. However, even when Wiener filtering is used, the peak of the correlation function has a width of several pixels, and the farther away from the peak, the more the correlation function consists of noise. To get a sub-pixel estimation of the actual translation, a small area centered on the peak is used to fit the peak of the correlation function.

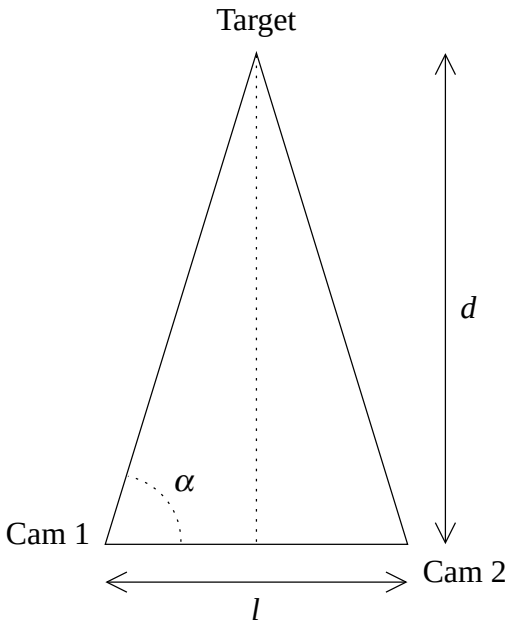

Figure 2. The two cameras of one measurement unit and the target location on the dome form an equilateral triangle. Given the distance l between the cameras and the distance d between the measurement unit and the target, the out-of-plane displacement can be calculated from the in-plane displacements measured by both cameras.

If such an algorithm would be used in a servo loop, for example, to do the correlation tracking for a tip-tilt mirror, the shape of the function used to fit the peak does not need to accurately resemble the shape of the correlation peak. As long as the algorithm always brings the image closer to the desired position, the accuracy can simply be improved by processing more images per second. In our situation we have no actuators, the accuracy of our estimate solely depends on choosing the right function for fitting the peak. The random pattern texture we currently use at the DOT results in correlation peaks that are cone-shaped. Using a parabolic function would result in discontinuities when the displacement in pixels is 0.5 away from the nearest integer. We currently implement the sub-pixel estimation in each direction separately. The two-dimensional cone shape then becomes a triangle in one dimension. When fitting a triangle to a 5-pixel-wide window of the measured correlation function $s(x)$, and given the position of the pixel with the highest correlation Δx, the sub-pixel estimate of the displacement $\Delta x_{\text{subpixel}}$ becomes:

$$\Delta x_{\text{subpixel}} = \Delta x + \frac{1}{2}\left[\frac{s(\Delta x-2)+s(\Delta x-1)-s(\Delta x+1)-s(\Delta x+2)}{s(\Delta x-2)-s(\Delta x-1)-s(\Delta x+1)+s(\Delta x+2)}\right] \tag{8}$$

We currently do the one-dimensional estimation in each direction to obtain the two-dimensional sub-pixel estimation.

3.8. Stereo decomposition

Our compact 3DD3 measurement units[2] contain two cameras, which look to the same target location via two inclined mirrors. The distance l between the two mirrors forms the baseline of the stereo system, which looks to the target at a distance d (see figure 2), where for simplicity the two cameras are thought to be directly looking to the target from the location of the two inclined mirrors. Given the measured displacements in the x and y direction by both cameras, the actual

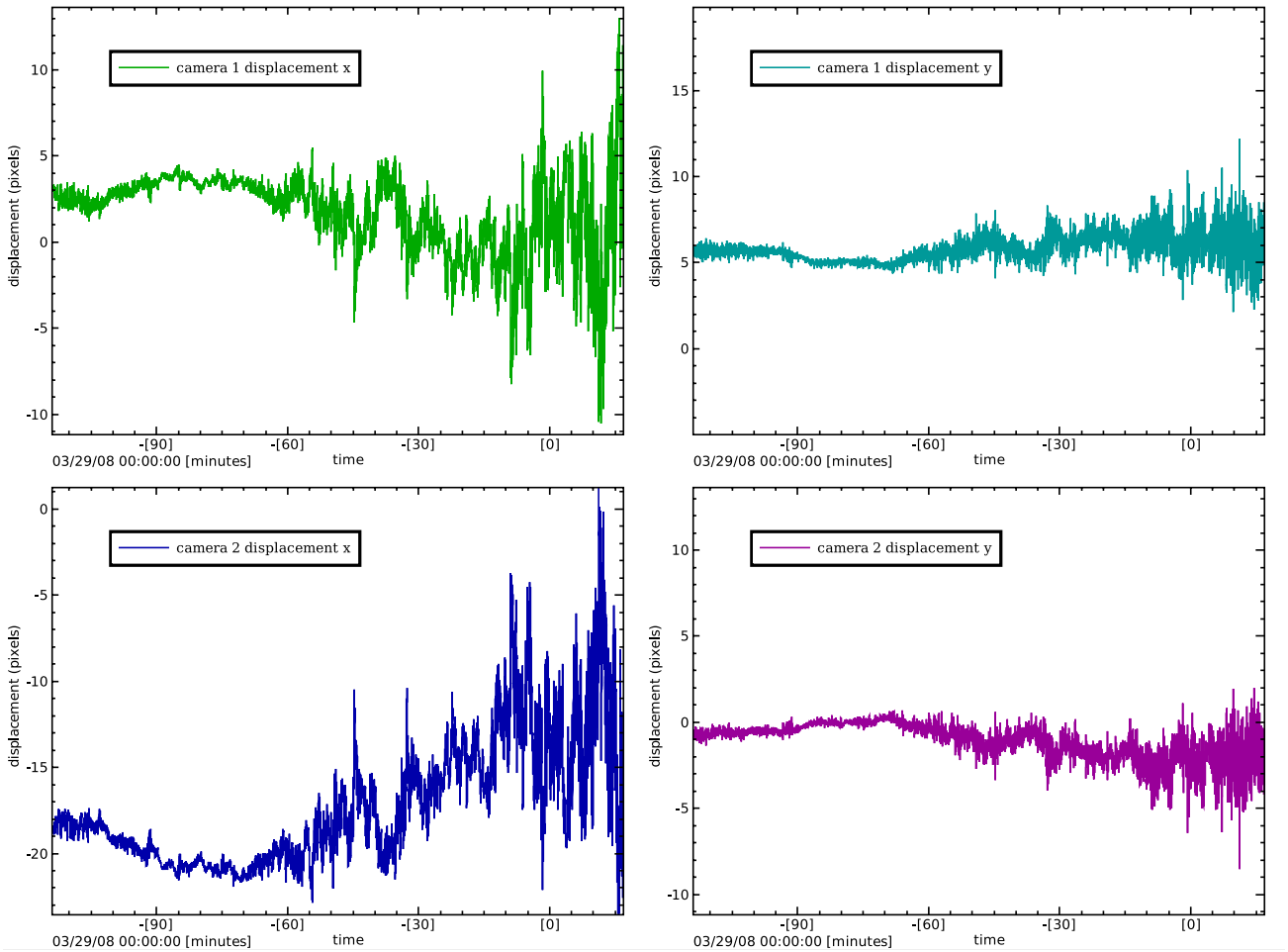

Figure 3. Raw displacement data from the two cameras of the same measurement unit. In the setup, one camera is rotated 180 degrees compared to the other, therefore the measured displacement of one camera has a sign different from that of the other camera.

displacement in three dimensions is given by the following equations:

$$\alpha = \tan^{-1}\frac{2d}{l} \tag{9}$$

$$\Delta x = \frac{1}{2}\frac{\Delta x_1 + \Delta x_2}{\sin\alpha} \tag{10}$$

$$\Delta y = \frac{1}{2}(\Delta y_1 + \Delta y_2) \tag{11}$$

$$\Delta z = \frac{1}{2}\frac{\Delta x_1 - \Delta x_2}{\cos\alpha} \tag{12}$$

If $d > 3l$, the term $\sin\alpha$ can be neglected, and $\cos\alpha \approx l/2d$. This means that the sensitivity of the measured displacement in the z direction is inversely proportional to the distance between the target and the measurement unit. In our current setup, where $l = 80$ cm (for our new mark 3 unit, $l = 100$ cm) and $d \geq 3$ m, the sensitivity of Δz is only of the order of a tenth of that of Δx.

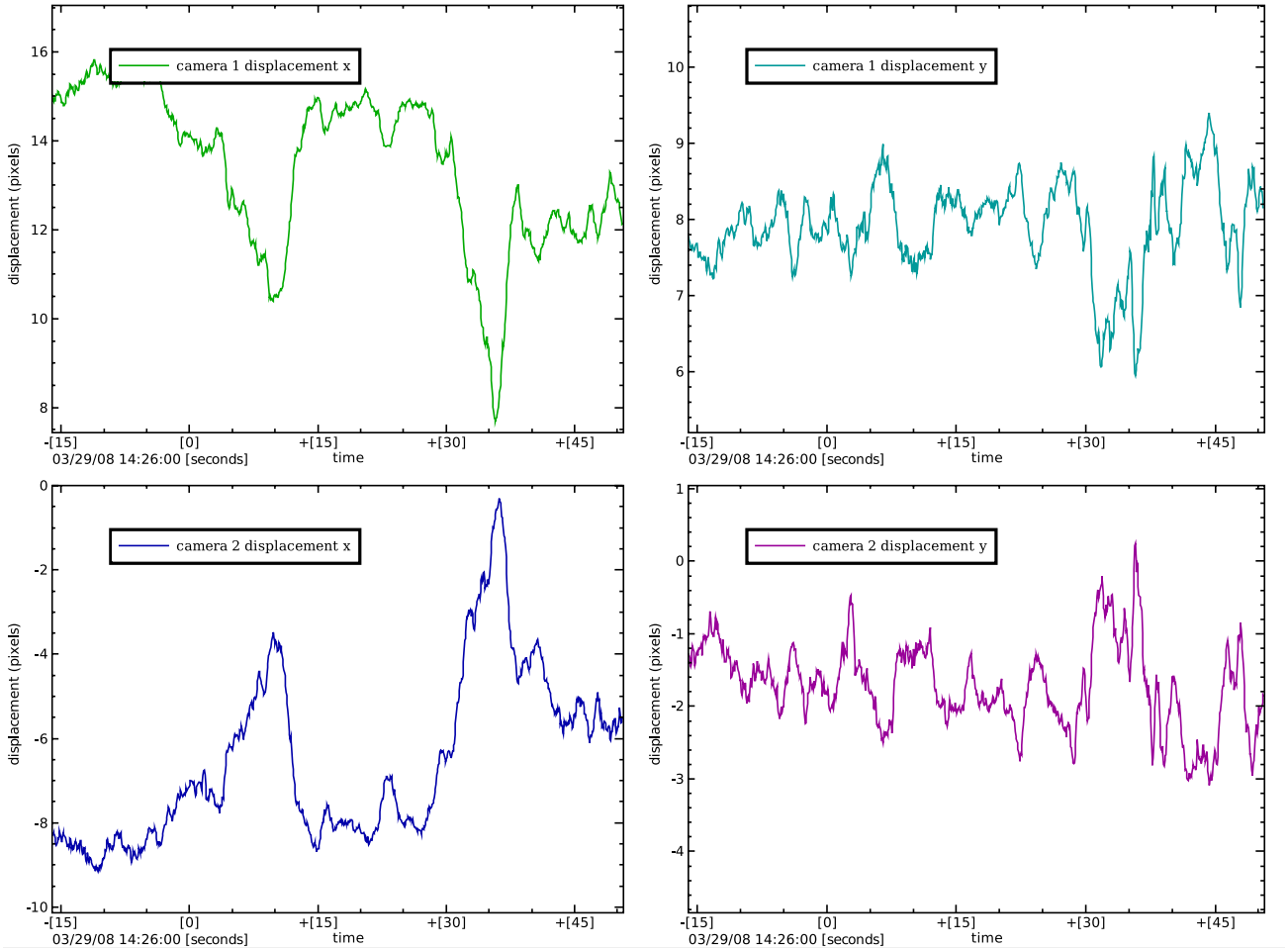

Figure 4. Detail of the raw displacement data from two cameras of the same measurement unit.

4. IMPLEMENTATION

The algorithm described in the previous section has been implemented in the C programming language. The highly optimized FFTW3 library[6] is used to perform the Fourier transformations. We use small form-factor computers with dual-core AMD64 processors running at a maximum of 2.8GHz. Four cameras are connected per computer. The image processing is CPU-bound, it is therefore important to make good use of the cache memory on the processor. We assign two cameras per CPU core, and run only one image processing thread per core. We pin each thread to a specific core to prevent cache invalidation when execution of a thread is moved to another core. Each thread maintains a queue of captured images from both cameras assigned to it. This ensures that only one FFT operation is running at a core simultaneously, again preventing unnecessary cache flushes. With this setup, one core is able to process 60 frames per second at 96% CPU load. Two cores together could in principle process the images from 4 cameras running at 30 fps, but the bandwidth of FireWire 400 bus limits us to approximately 95 fps. At some extra cost an extra FireWire interface card could be added to the computers, thereby removing the bandwidth limit.

5. RESULTS

A number of measurement units have already been installed on the Dutch Open Telescope and are in operation. The raw results from two hours of measurements of a single unit is plotted in figure 3, and a more detailed graph of the results on a timescale of one minute is plotted in figure 4. Since one of the cameras in a measurement unit is rotated 180 degrees

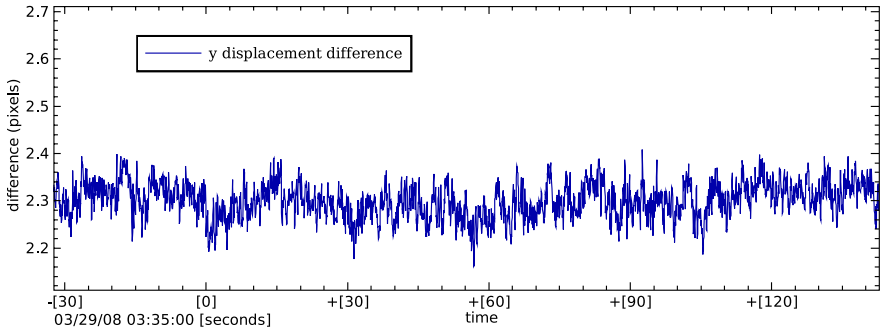

Figure 5. The *y* displacement of both cameras of the same measurement unit should be the same. We can make an estimate of the noise in the measurements by subtracting the measured *y* displacements. Due to daily temperature variations, alignment of the cameras can shift slightly during the day, which can result in a difference in measured displacement of several pixels. The short-term noise however has an RMS value of 0.09 pixels.

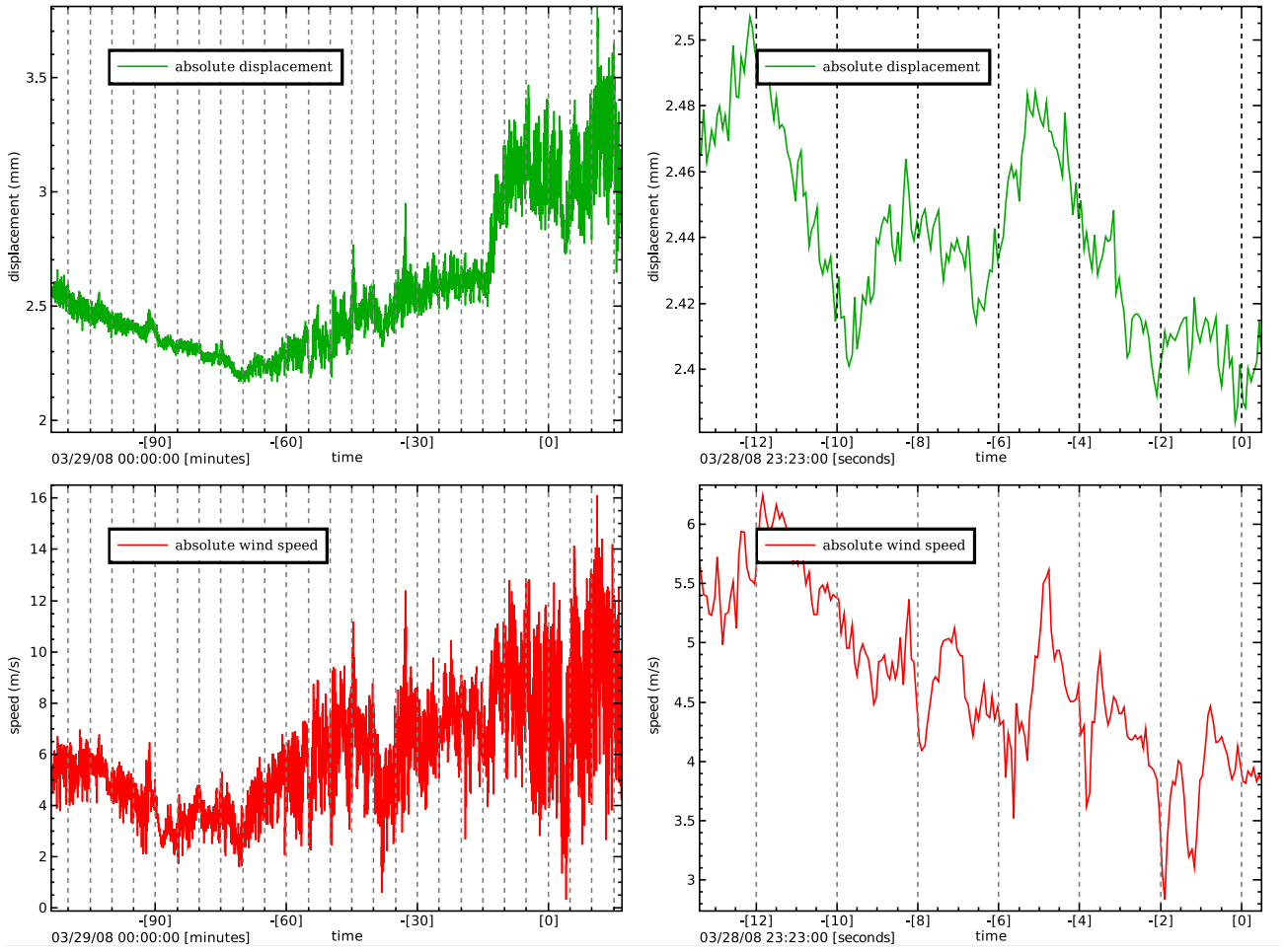

Figure 6. Comparison of the wind speed outside the dome, as measured by an ultrasonic anemometer, and the measured displacement from one of the measurement units. Although displacement is not a simple linear function of wind speed, the graphs show a clear correlation between wind speed and displacement, at both small and large timescales.

compared to the other, the raw results show a difference in sign between both cameras in the measured displacement. Both cameras work independently, but are looking at the same point on the bow. As expected, the measured displacement is therefore very similar. The absolute value of the displacement is not the exact opposite though; there is a slight mechanical drift due to temperature variations during the day, and the reference images for each camera were calculated independently. This can cause the displacement as measured by both cameras to drift slowly over time, which has to be taken care of in data post-processing.

Although there can be a difference in the measured displacement in the *x* direction due to out-of-plane movement of the target, the measured displacement in the *y* direction should be the same for both cameras. The difference in *y* displacement, as shown in figure 5 gives us information about drift and noise. The long-term drift due to daily temperature variations in the DOT is in the order of pixels. The short-term variations are caused by noise in the read-out of the CCDs and the limitations of the image processing algorithm. The root mean square (RMS) of the short-term variations for the setup is 0.09 pixel. Given the optical magnification of $1:10$ and the fact that the pixels of our cameras are 5.6×5.6 μm, the effective sensitivity of our measurement unit is 5.04 μm.

6. CONCLUSION

Using relatively cheap, off-the-shelf cameras and computer components, it is possible to continuously measure displacement of a point on a surface at a distance of more than 3 meters, in three dimensions, at 30 frames per second, with a sensitivity of 5 μm (in-plane). The measurement units are currently used to measure the deformation of the domes of the DOT and GREGOR telescopes, but could also be used to measure displacement of other objects. It is possible that the sensitivity can be made even better in the future by improving the algorithm used to determine the displacement.

7. ACKNOWLEDGEMENTS

The Technology Foundation STW in the Netherlands supports this work financially under grant 06640. Technical support comes from the GREGOR and DOT groups. The GREGOR is operated by the German consortium of the Kiepenheuer Institut für Sonnenphysik, the Astrophysikalisches Institut Potsdam, the Institut für Astrophysik Göttingen and other national and international partners at the Observatorio del Teide (OT) on Tenerife. The DOT is operated by Utrecht University at the Observatorio del Roque de los Muchachos (ORM) on La Palma. Both observatories are operated by the Instituto de Astrofsica de Canarias in Spain. The DOT has been built by instrumentation groups of Utrecht University, the Central Workshop of Delft University (now DEMO-TU-Delft) and several firms with specialized tasks with funding from the Technology Foundation STW. The DOT team enjoys hospitality at the solar telescope building at ORM of the Royal Swedish Academy of Sciences.

REFERENCES

1. F.C.M. Bettonvil, R.H. Hammerschlag, A.P.L. Jägers and G. Sliepen, "Large fully retractable telescope enclosures still closable in strong wind", Astronomical Instrumentation, Proc. SPIE **7018-60**, 2008.
2. A.P.L. Jägers, G. Sliepen, F.C.M. Bettonvil and R.H. Hammerschlag, "Fast foldable tent domes", Astronomical Instrumentation, Proc. SPIE **7018-64**, 2008.
3. W.H. Press, S.A. Teukolsky, W.T. Vetterling and B.P. Flannery, *Numerical Recipes in C*, p. 547-549, Cambridge University Press, Cambridge, 1992.
4. Dutch Open Telescope, `http://dot.astro.uu.nl/`.
5. GREGOR telescope, `http://http://gregor.kis.uni-freiburg.de/`.
6. FFTW Fourier transformation library, `http://www.fftw.org/`.