

Brief explanatory example of a practical approach to UAV/drone camera calibration using very low-level aerial photography and the *CameraCalibrator*

Introductory Comment

Cameras used on UAVs/drones can be calibrated using images gathered either during a flight mission or 'on the ground' with the camera either attached or unattached to the drone. The calibration from near-nadir aerial photography is not always practical, since the use of ground control points and camera station GPS positions are generally warranted. It can also be impractical to try to calibrate a camera affixed to a hand-held (i.e. not in flight) drone where either the size of the drone or manoeuvrability of the integrated camera does not allow positioning/pointing of the drone camera to form the desired network geometry of convergent imaging angles, mixed scale images and camera roll diversity.

Here, a very practical alternative approach to robust and accurate drone camera calibration is described. This entails the recording of a network of very-low-level (a few metres above ground) aerial images of an array of Photometrix coded targets. In essence, the procedure mimics what is done when an indoor calibration is performed with the drone (or camera alone) being hand-held.

The steps of the calibration procedure will be outlined by way of a practical example, so this document can be used as a tutorial. For a comprehensive, general description of automatic camera calibration, users are referred to Section C2 of Part 1 of the *CameraCalibrator* User Manual.

1) Target Array: Lay down an array of 40 or so Photometrix codes, either red codes (preferable) or printable white-on-black codes, such that the area covered is around 2.5 x 2.5 m (8 x 8 ft) as indicated in Figure 1 below. The following aspects are important:

- The codes must be flat and they must not move at all during photography, so having them taped down or mounted on a heavy enough backing so they are not moved by UAV downdraft is essential.
- The codes must not all lie in the same plane; at least 3 – 4 must be out of plane by 40 – 50cm or so.
- Generally speaking, the standard-sized 8mm red codes are quite suitable for camera-to-object distances up to 3m (10 ft). For greater camera-to-object distances, it may be necessary to utilise codes with larger dots, such as printable white-on-black codes.

2) Photography: The images of the codes will be read automatically by *CameraCalibrator*, so it is imperative that the photography is captured such that the codes are high contrast and in clear focus, as indicated in Figure 1 (recall that the focal length needs to be fixed for all images, generally at infinity focus for subsequent drone mapping projects). It is important to conduct

some initial test photography to ensure satisfactory image exposure is being obtained. Helpful hints include:

- Where possible, stop-down the lens to a small aperture (e.g. f/11 or f/16) to optimise focus via an increased depth of field.
- Underexposing the photography by 1 to 2 stops often enhances the contrast between code 'dots' and their background, thus making automatic image scanning faster and more accurate.
- Minimize image blur by ensuring the drone is steady when each image is captured.

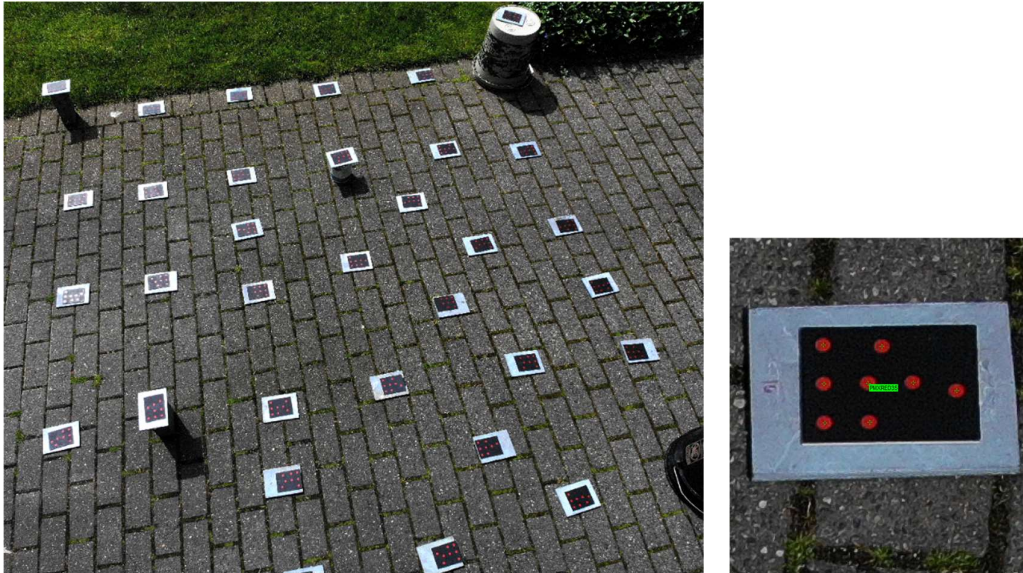


Figure 1

3) Camera Station Geometry: There is no set required network geometry, instead the imaging configuration needs to exhibit a number of features, noting that it must also meet the photographic quality requirements mentioned above. An example network, and one which will be used throughout this explanatory note, is shown in Figure 2.

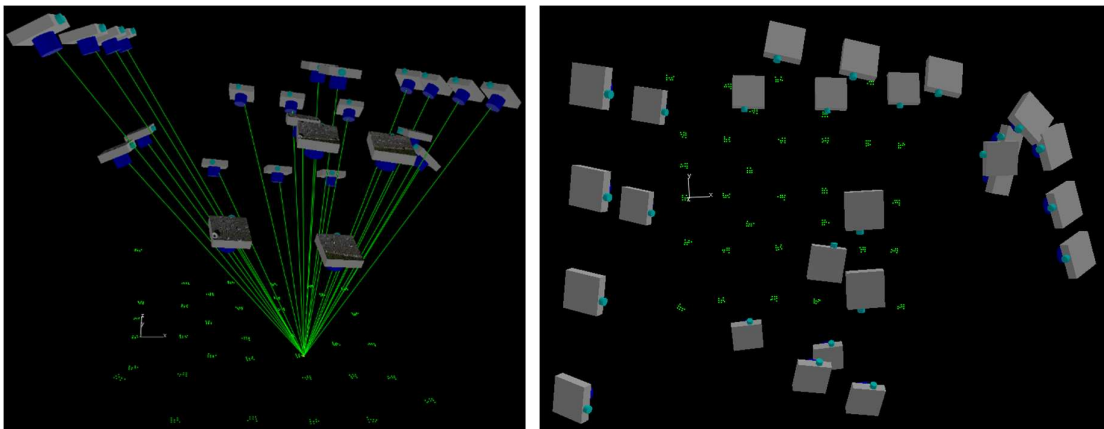


Figure 2

This network displays the following required features:

- A convergent imaging configuration, whereby the optical axis is tilted away from the vertical by 30 (minimum) to 45 degrees. Not all images need to be convergent, just the majority.
- It is highly desirable to incorporate a mix of image scales (i.e. use of varied camera-to-object distances) within the photography. Note in Figure 2 how this is achieved through the use of three different height levels; substantial image scale variation is a very desirable attribute.
- It is important that all areas of the image format are covered, not necessarily within a single image, but across all images. Thus, it is not generally advisable to point the camera to the centre of the target array for all images. And, it is desirable, but certainly not necessary that all codes appear in every image, though each should appear in a minimum of 4 – 6 images.
- In general, the area covered by the target array should be such that it is slightly bigger than the area covered by a single photo at maximum camera-to-object distance. In instances where images cover a significantly greater area than the target array, either fly the drone lower, or increase the size of the target field.
- As regards the number of images, there is no maximum number, though the minimum needed to achieve all required attributes is generally about 20. Image measurement is automatic, so capturing 40 – 50 images, for example, will have minimal impact on processing time.
- There needs to be a diversity of camera roll angles within the network. This is not always achievable with oblique imagery captured from a drone, since it's generally not possible to rotate the camera around its optical, pointing axis. However, orthogonal rolls can be achieved by adding near-nadir images, rotating the drone by 90 and 180 degrees between images. To achieve the required orthogonal roll angles, the drone can be hovered over the codes with the camera pointing downwards (near-nadir). Two or more images are then recorded, with the drone's yaw rotation changing by 90 degrees for each image, as indicated in Figure 3.

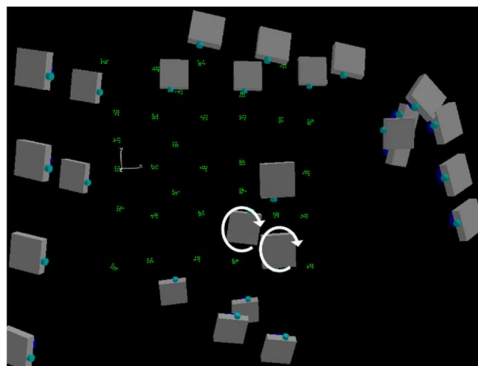


Figure 3

The 26-image, 288-point (36 codes) network shown in Figure 2 was recorded with the integrated camera of a DJI Mavic Air drone. This camera has an image resolution of 4056 x 3040 1.6 micrometre pixels and a nominal focal length of 4.8mm, these values being automatically read from the EXIF header of the jpeg images by *CameraCalibrator*. The heights above ground of the camera stations ranged from 2.2m to 3.9m (7 to 13ft) for the oblique/convergent images, to almost 4m for the three near-nadir, downward-looking images. At a setback distance of 4m, the imaging scale is 1: 830 (flying height divided by focal length), and at this scale the size of the 8mm diameter code dots in the images will be just under 10 micrometres, or 6 pixels. A target size in the images of greater than 3-4 pixels is required for accurate measurement.

4) *CameraCalibrator* Data Processing

- i) Initiate a *CameraCalibrator* project and import the images, after which the target type should be selected from **Edit|Project Settings|Targets** (**Red** would be appropriate for this example).
- ii) Next, the user may choose to set a unique ID for the camera, by double-clicking on the camera icon and selecting **Add unique ID**.
- iii) An optimization of the image scanning parameters is the next step. This is initiated by opening an image and selecting the **Q-key**. This brings up the Image Scanning and Auto-Measurement Settings dialog. The **Optimize** button is then selected to determine the best scanning threshold, as indicated in Figure 4. Note that this process may take a few minutes, depending upon the number of images and the general exposure of the images (the process is faster with under-exposed images). Click **OK**.

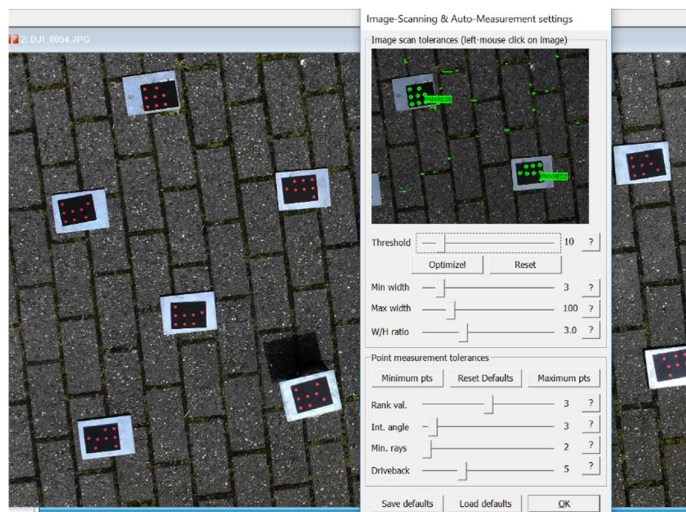


Figure 4

- iv) The *CameraCalibrator* automatic calibration computation is then initiated by selecting **Photogrammetry** on the toolbar and **Autocal** from the pull-down menu. The dialog

shown in Figure 5a is then displayed. In the case of a first-time calibration, this will indicate the nominal focal length and zero values for other calibration parameters. Non-zero values are likely if a camera of the same make/model is already in the camera database and has been previously calibrated. **Begin** initiates the image scanning and self-calibrating bundle adjustment operations, which terminate with the results being presented as shown in Figure 5b.

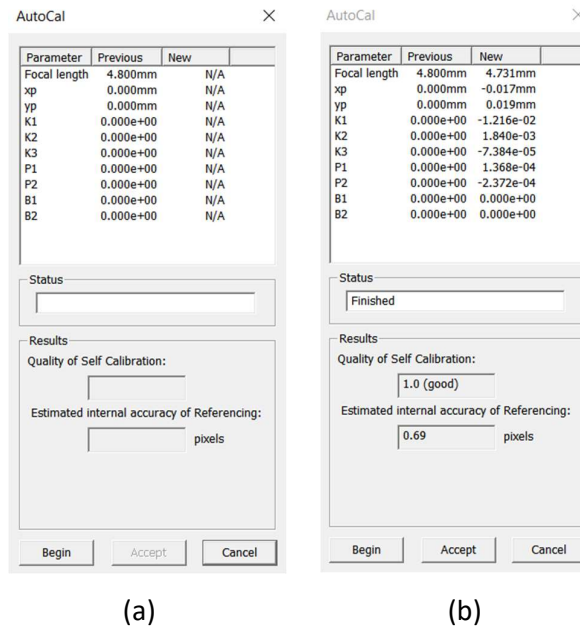


Figure 5

The quality of the camera calibration is indicated by the **Estimated internal accuracy of referencing**, and by the **Quality of Self Calibration**. The former of these will generally range from about 0.3 pixels for a good quality DSLR or mirror-less camera up to 0.8 pixels for a small consumer-grade drone with an integrated camera, such as the DJI Mavic Air or Phantom series. The **Quality of Self Calibration** has an optimal value of 1.0, and values to 1.5 are acceptable. Values higher than 1.5 indicate a weak network geometry and thus sub-optimal determination of camera parameters.

Accept is selected to complete the calibration process; the camera parameters are written to the *CameraCalibrator* camera database, and the project can be saved (*project name.clx*).

5) Output of Calibration Report and a note on Lens Distortion: To output a calibration report in pdf format, double-click on the camera icon and select **Export PDF Report**. The report lists the parameters and their estimated accuracy, and it also shows plots of radial and decentring distortion, the former of which is the most significant. These same plots can be shown on the screen again by double-clicking on the camera icon and selecting **Show Distortion Curves**. Note the red vertical line in the distortion plot of Figure 6. This indicates the maximum radial distance to a target point within the set of images, and the red line should be as far to the

right as possible, since the dotted line indicates an extrapolated section of the curve, which is therefore of lower reliability.

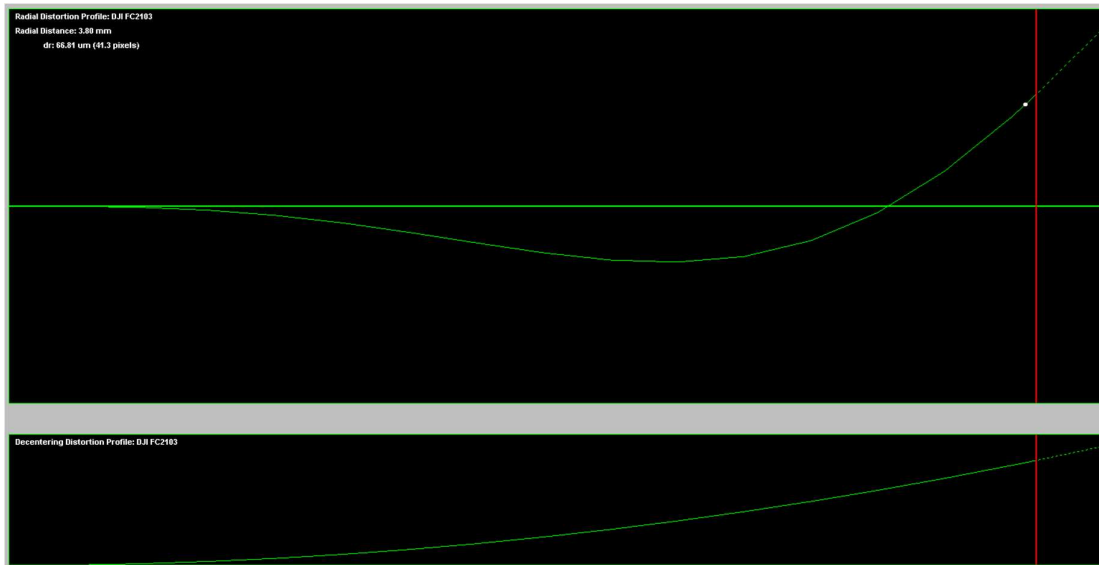


Figure 6

FINAL NOTE: As detailed in the *CameraCalibrator* User Manual, an alternative means to self-calibrate a camera is via the AutoMeasure (**R++**) procedure. This process affords the user flexibility over the choice of calibration parameters and experienced *CameraCalibrator* users may well find the **R++** option the more comprehensive calibration route. Also, in this user guide, calibration via coded targets has been described. However, it is also possible in favourable circumstances to perform the calibration via a targetless approach, as explained in the User Manual. For very low-level aerial UAV camera calibration networks, users will likely find the use of targets to be a more practical and straightforward option than the targetless approach.