

Algorithm for anonymous markers, reprojection filter, minimum number of frames and criterion for a successful calibration

From an anonymous T-wand to the minimum achievable error

SIMONE TORTORELLA

October 2026

ABSTRACT

We describe and evaluate the algorithm that calibrates 8 IR monochrome cameras from anonymous blobs: T recognition by affine fit with ambiguity rejection, initial estimate from the terna, sparse bundle adjustment with rejection of observations with a high normalised residual, multi-view relabelling by Hungarian assignment, final bundle adjustment. Labelling takes 0.08 ms per image; with an ambiguity threshold of 2 it leaves 1.4% of images wrong, which relabelling reduces below 0.1% of the observations. Rejection at $k = 2.5$ –3 standard deviations is a good compromise. About 100–200 frames are enough if spread over at least 20 s of motion: duration (volume coverage) matters more than the number of frames; the bundle adjustment with 400 frames and 8 cameras takes about 15 s in Python. The reprojection residual tells good calibrations from wrong ones poorly; a validation wand (frames kept aside) does better: in 90 sessions with random defects the rule wand deviation ≤ 0.75 mm, mean deviation ≤ 0.10 mm and coverage $\geq 5\%$ never accepts calibrations with a p95 beyond 2.9 mm, while a rule on the residual accepts some up to 10.6 mm. The error budget points to a minimum achievable error of about 0.10 mm (p95 in the covered volume, with an unmodelled centroid error of 0.02 px) and shows what limits it: the unmodelled centroid error and the terna tolerance, not blob noise.

Keywords: bundle adjustment · unlabelled markers · reprojection · outlier rejection · calibration validation · error budget

1. Introduction

Paper 1 fixed the object (asymmetric 4-marker T, 3D terna) and Paper 2 the number and placement of cameras. Here we describe the algorithm that turns anonymous blobs into a calibration, choose its thresholds using the simulation's known truth, measure what it costs in frames and time, and look for a criterion, computable without the truth, to say whether a calibration worked. Finally we build an error budget to see which minimum error can be reached and what limits it.

2. The algorithm

Algorithm 1 — calibration from anonymous blobs

1. **Blobs**: threshold at 40% of white, connected components, weighted centroid, diameter and circularity of every blob.
 2. **Terna**: average of 30 frames; PnP for every camera with nominal intrinsics → initial estimate.
 3. **Per-image labelling**: for every subset of 4 blobs (also dropping one as a reflection) and every permutation, affine fit to the T drawing; keep the best if $r^* < \tau$ and every different labelling has $r > \text{amb} \cdot r^*$, otherwise reject the image.
 4. **Shape filter**: discard blobs that are not round or smaller than the other markers of the same image suggest (partial occlusion).
 5. Sparse **bundle adjustment** (intrinsics, extrinsics, T positions, delays) with a rigidity constraint.
 6. **Rejection**: remove observations with normalised residual $e > k$; repeat the bundle adjustment (2 times).
 7. **Multi-view relabelling**: project the T estimated from all cameras into every image; assign *all* blobs (also of rejected images) with the Hungarian algorithm within g px.
 8. **Final bundle adjustment** with rejection; **validation** with the frames kept aside (§6).
-

The normalised residual of an observation $\mathbf{u}_{ckm}$ (camera c , frame k , marker m) is

$$e_{ckm} = \frac{\|\pi_c(\mathbf{X}_{km} + \delta_c \mathbf{V}_{km}) - \mathbf{u}_{ckm}\|}{\sigma_{ckm}}, \quad \sigma_{ckm}^2 = \sigma_{\text{blob}}^2(d) + \sigma_0^2 \quad (1)$$

where π_c is the projection with distortion, δ_c the camera delay, $\mathbf{V}$ the marker velocity and $\sigma_{\text{blob}}(d)$ the expected centroid error for a blob of diameter d (Paper 1). Relabelling solves, for every image, the assignment

$$\min_{\phi} \sum_i \|\pi_c(\hat{\mathbf{X}}_{k,\phi(i)}) - \mathbf{u}_{ci}\| \quad \text{s.t. each marker used once, } \|\cdot\| < g \quad (2)$$

3. Labelling: thresholds and speed

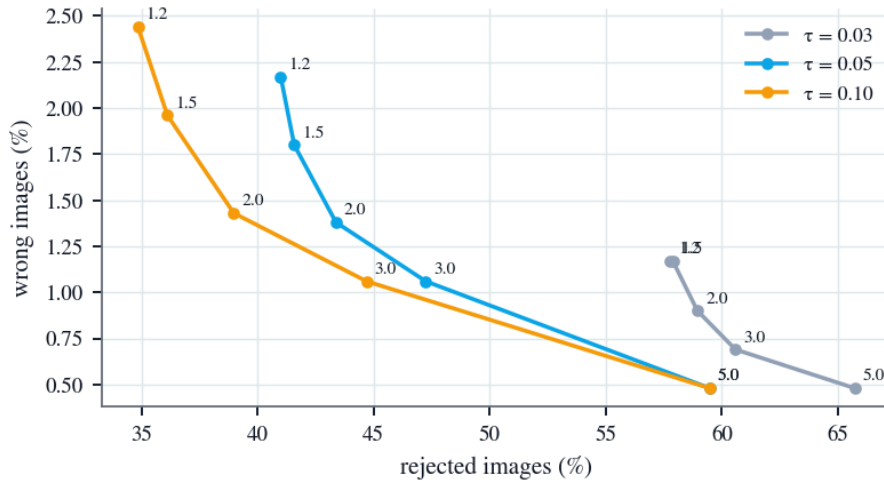


Figure 1. Trade-off between wrong and rejected images for the ambiguity threshold amb (numbers on the points) and the absolute threshold τ (T4a, 10% reflections, 8 cameras, 2400 images).

Comment. A stricter ambiguity threshold reduces errors and increases rejections; τ shifts the curve. With $\text{amb} = 2$ and $\tau = 0.05$, 1.4% of the images are wrong and 43% rejected. There is no need to push further: rejected images are recovered by relabelling and residual errors fall below 0.1% of the observations (Paper 1). Labelling takes 0.08 ms per image in vectorised Python: 8 cameras at 60 fps are manageable in real time even without optimisation.

4. Reprojection and filtering threshold

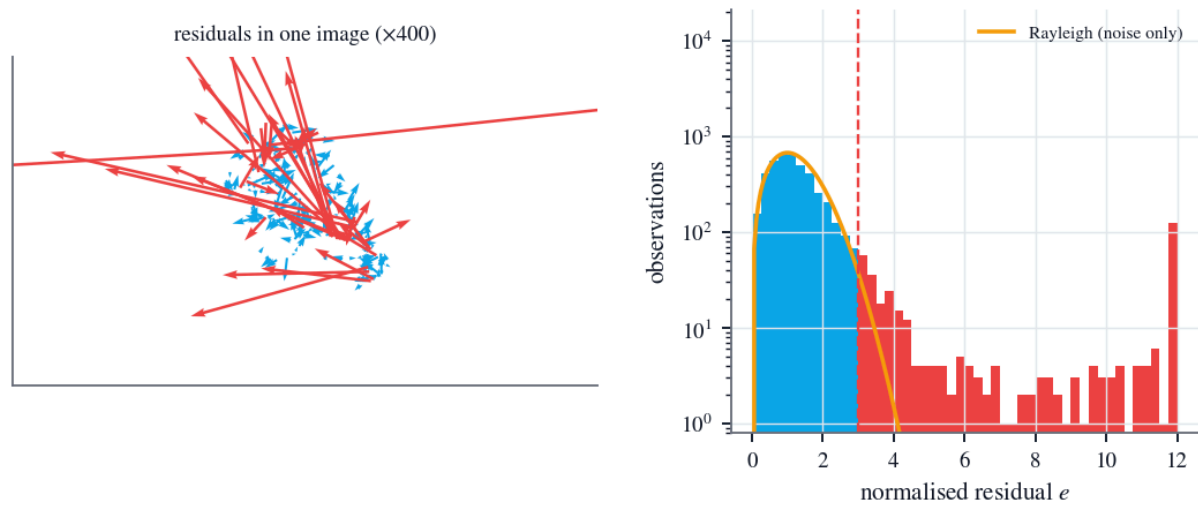


Figure 2. Left: residual vectors (magnified 400 times) in one image, against the final solution; red: normalised residual beyond 3. Right: histogram of the normalised residuals of all observations (rejected ones included) of a session with 0.4% of the images containing two swapped markers; yellow: the distribution expected with Gaussian noise only (Rayleigh); the last bar collects residuals beyond 12.

Comment. With Gaussian noise only the normalised residual follows a Rayleigh distribution: 1.1% would fall beyond 3. Here 8.4% falls beyond 3: besides the noise tails there are partially hidden markers (the centroid is shifted by tenths of a pixel, Paper 1) and, in the last bar, the swapped markers, tens of standard deviations away. The k -sigma filter therefore removes two different things: gross identification errors and observations distorted by partial occlusion that the shape filter missed.

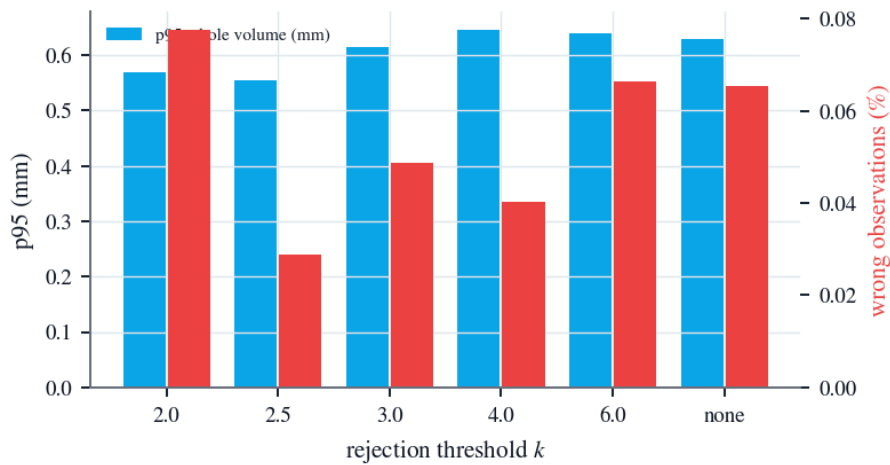


Figure 3. Full algorithm (labelling, rejection, relabelling) versus the rejection threshold k : final error and share of wrong observations left.

Comment. After relabelling wrong observations are so few (0.03–0.07%) that the threshold changes the final error little (0.57 mm at $k = 2$, 0.63 mm without rejection). Too strict thresholds ($k = 2$) throw away 17% of the good observations (Rayleigh tails). We choose $k = 2.5$ –3: it removes gross errors before relabelling, when they are more numerous, and costs 3–8% of the data.

5. How many frames and how long

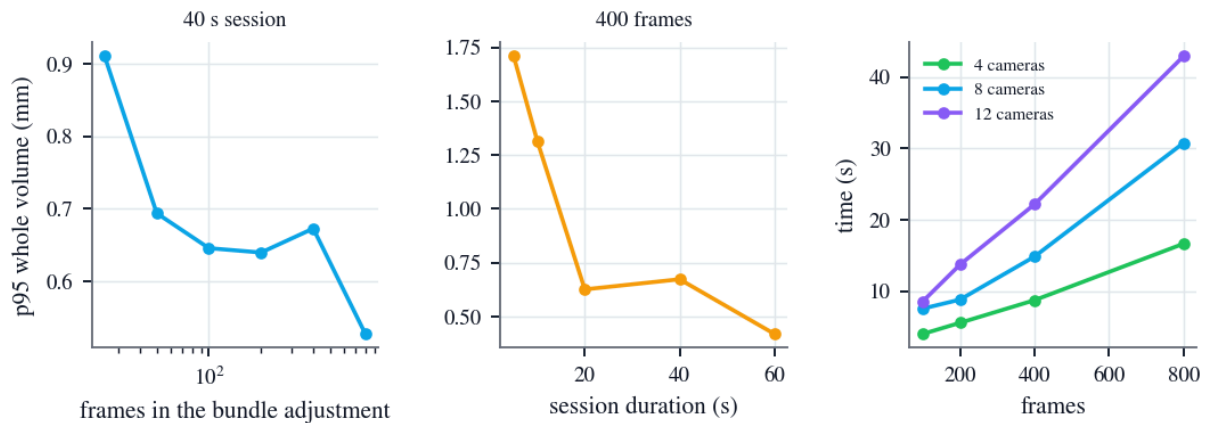


Figure 4. Left: error versus number of frames used (40 s session). Centre: versus session duration (400 frames). Right: bundle adjustment time (3 passes, Python + SciPy, one core).

Comment. With 25 frames the error is 0.91 mm; from 100 frames on it stabilises (0.65 mm with 100, 0.53 mm with 800). Duration matters more: with the same 400 frames, 5 s give 1.71 mm, 20 s 0.63 mm, 60 s 0.42 mm, because a long session covers more volume. **Minimum: 100–200 frames spread over at least 20 s, better 40–60 s.** Time grows linearly with frames and cameras: 8 cameras and 400 frames take about 15 s; it is not a bottleneck, and a C++ implementation (Ceres) would be 10–50 times faster.

6. When the calibration has worked

In a real installation there is no truth. We need computable quantities that predict the error. We simulated 90 sessions with random defects (few frames, short sessions, a person leaving the volume, initial focal length wrong by up to 30%, an imprecise terna, unmodelled error up to 0.3 px, up to 5% of images with swapped markers) and computed for each: the RMS reprojection residual, the **coverage** (share of the volume within 30 cm of a T position seen by ≥ 2 cameras), the minimum number of observations per camera and a **validation wand**: one frame in five does not enter the bundle adjustment and is triangulated with the estimated cameras; marker distances are compared with the known ones of the T (mean and standard deviation of the deviation). A session is "good" if the p95 in the covered volume is ≤ 1 mm.

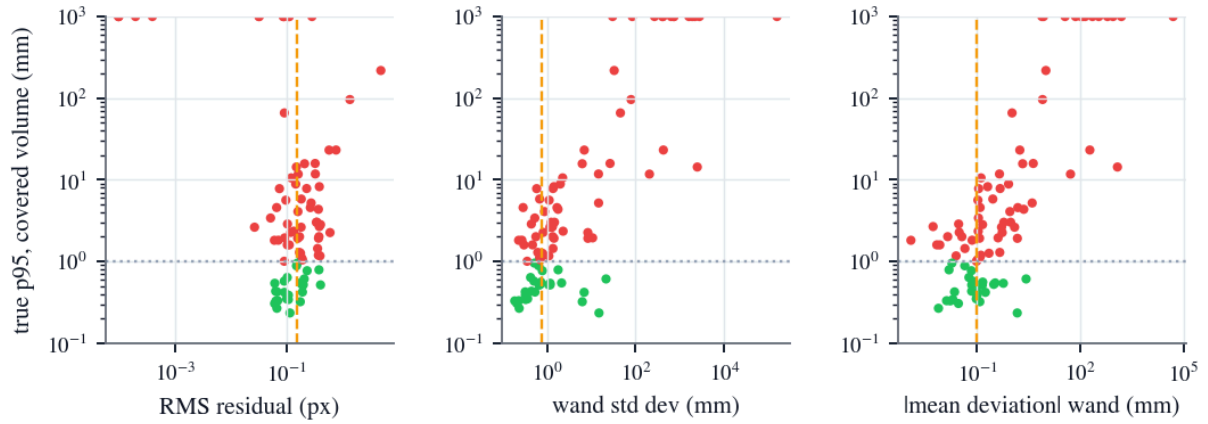


Figure 5. 90 sessions with random defects: true p95 (covered volume) versus three diagnostics. Green: good (≤ 1 mm); red: not. Yellow dashed: rule thresholds.

Comment. No diagnostic separates the two classes perfectly, but some do much better than others. The reprojection residual is the worst (AUC 0.60: 0.5 would be chance, 1 perfect separation): with little data the bundle adjustment fits cameras and points to the observations and the residual stays small even when the calibration is wrong. The validation wand is the best (AUC 0.76 for the standard deviation, 0.76 for the mean deviation), followed by the number of observations per camera (0.73).

Table 1. The two rules on the 90 sessions (27 good).

Rule	accepted	of which good	maximum p95 among accepted
wand: dev. ≤ 0.75 mm, $ \text{mean} \leq 0.10$ mm, coverage $\geq 5\%$	20	13	2.9 mm
residual ≤ 0.15 px, coverage $\geq 5\%$	29	14	10.6 mm

The wand rule also accepts a few calibrations above 1 mm, but **never beyond 2.9 mm**; the residual rule accepts some up to 10.6 mm. In practice the wand should be used as an error estimate rather than a yes/no test: the 95th percentile of the wand deviations has a correlation of 0.70 (log scale) with the true error, the residual 0.49. An independent check is even better: a second wand of known length moved through the whole volume after calibration, because the validation frames here come from the same trajectory as the calibration and do not see the regions the T never crossed.

7. Error budget and feasibility

To see which minimum error can be reached we start from the reference configuration (8 Full HD cameras at two heights, 60 fps, 20 s, 30 mm markers, hardware trigger) and remove or improve one error source at a time.

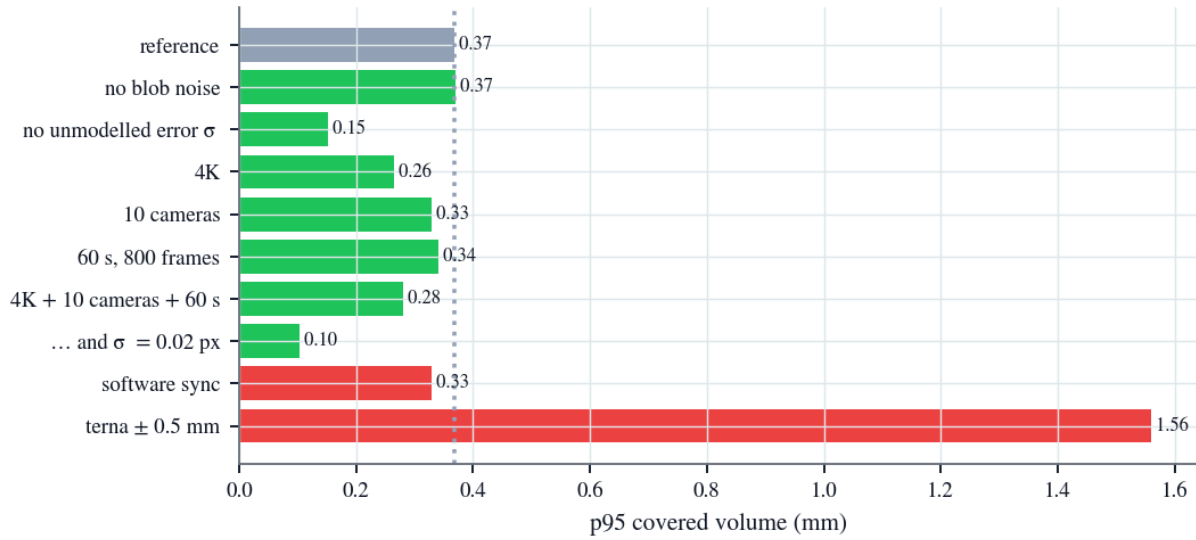


Figure 6. Error budget: p95 in the covered volume removing or improving one source at a time (median of 4 sessions). Green: improvements; red: degradations.

Comment. Blob noise (Paper 1 model) is irrelevant: removing it changes nothing (0.37 against 0.37 mm). The unmodelled error σ_0 matters: without it the error drops to 0.15 mm. 4K, 10 cameras and 60 s sessions help little one by one (0.26, 0.33, 0.34 mm); all together they lead to 0.28 mm with $\sigma_0 = 0.08$ px and to ****0.10 mm** with $\sigma_0 = 0.02$ px, the minimum achievable in this scenario** (p95 in the covered volume; 0.14 mm over the whole volume). Two sources instead ruin it: software synchronisation (0.33 mm in the covered volume, 3.01 mm over the whole volume) and the terna tolerance (1.56 mm with ± 0.5 mm).

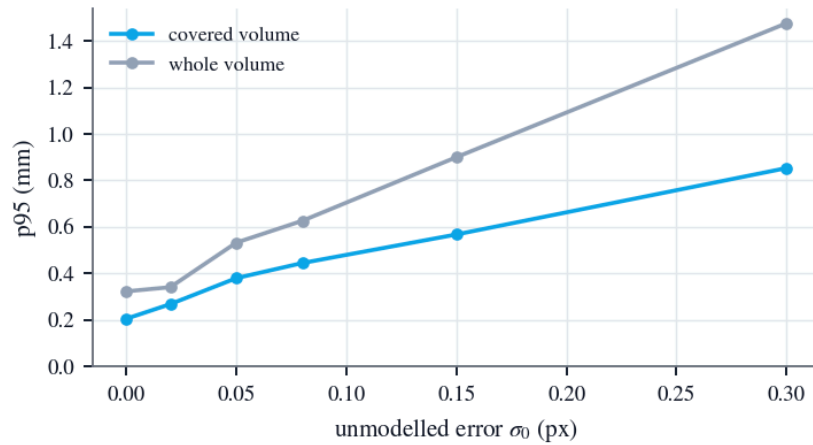


Figure 7. Sensitivity to the unmodelled centroid error σ_0 (8 Full HD cameras).

Comment. The error grows almost linearly with σ_0 : from 0.27 mm at 0.02 px to 0.85 mm at 0.3 px. It is the first parameter to measure on a prototype, filming a still sphere for a few minutes: the standard deviation of the centroid, after removing slow drifts, is a direct estimate of σ_0 .

Table 2. Feasibility: what limits the error and what can be done to lower it further.

Parameter	Effect	How to lower it
Unmodelled error σ_0	dominant; $\sim$ linear	richer distortion model, global shutter, rigid mounts, thermal stabilisation
Terna tolerance	dominant on absolute accuracy	terna machined to 0.1 mm or measured with a CMM arm
Synchronisation	$\times 4$ –6 over the volume	hardware trigger
Partial occlusion	bias $\propto$ diameter	small markers, shape filter, 3σ rejection
Volume coverage	error outside the covered volume	40–60 s sessions guided by a live coverage map
Resolution / number of cameras	secondary for calibration	4K if markers must be small

8. Conclusions

Algorithm, thresholds, minimum error

- Labelling: affine fit over every assignment, $\text{amb} = 2$, $\tau = 0.05$; no tracking (it propagates errors).
- Filters: shape (circularity ≥ 0.93 , diameter $\geq 0.9 \times$ the expected one) and rejection at $k = 2.5$ –3 of the normalised residual.
- Multi-view relabelling within 6 px (Full HD) before the final bundle adjustment.
- Session: ≥ 20 s (better 40–60 s), 100–400 frames; T seen by ≥ 2 –3 cameras together.
- Criterion: validation wand with deviation ≤ 0.75 mm and mean deviation ≤ 0.10 mm, coverage $\geq 5\%$ (never accepted calibrations beyond 2.9 mm); better still, a second wand moved through the whole volume. The reprojection residual alone is not enough.
- Minimum achievable error in simulation ~ 0.10 mm (p95, covered volume); in practice limited by σ_0 and the terna.

Limits: thresholds were chosen on simulated data with known truth; on a real system they must be checked with an independent measuring object (for example a second wand of known length). The 90 sessions of the criterion cover plausible defects but not all (for example cameras knocked during the session).

References

1. B. Triggs, P. McLauchlan, R. Hartley, A. Fitzgibbon, *Bundle adjustment — a modern synthesis*, LNCS 1883, 2000.
2. H. W. Kuhn, *The Hungarian method for the assignment problem*, Naval Research Logistics Quarterly 2, 1955.
3. R. Hartley, A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed., 2004.
4. M. A. Fischler, R. C. Bolles, *Random sample consensus*, Communications of the ACM 24(6), 1981.
5. W. Kabsch, *A solution for the best rotation to relate two sets of vectors*, Acta Crystallographica A32, 1976.