**MoRe-T2: An easy-to-use, low cost tracking system for mobility research**

Chinemelu Ezeh, Catherine Holloway, Tom Carlson
University College London, London, UK
chinemelu.ezeh.14@ucl.ac.uk
c.holloway@ucl.ac.uk
t.carlson@ucl.ac.uk

## Abstract

In order to achieve inclusive access for all, it is important to measure how difficult access is for people, especially those with mobility impairments. One practical and effective way of doing this is to track how they move around in a given environment. However, systems for tracking people are often costly and complicated to use. This article concerns the development and application of a low cost computer vision-based tracking system called MoRe-T2 (mobility research trajectory tracker), which tracks the position and orientation of QR-code-like markers. These markers can be simply printed onto paper and attached to people or assistive technologies, such as wheelchairs and scooters. The system can then be used to quickly characterize the trajectories people make under different conditions (e.g. when using a wheelchair, exiting through a train's sliding doors, approaching a ramp or moving through crowded environments etc.). This adds a level of information (orientation) and the ability to uniquely identify many more participants, which is not available using more standard video tracking techniques such as 'blob' detection (i.e. simply tracking a blob of a certain colour for example). The system works by first attaching the uniquely identifiable paper-printed markers to the participants or their assistive technologies. A system of time-synchronized cameras is set up around the experiment area, such that at any given time every marker is visible to at least one camera. The cameras' positions can be calibrated automatically, in a simple process, using a single marker. Using these known positions, videos of the scene are recorded and post-processed with the help of the ARToolkitPlus library, which is used to calculate the position and orientation of markers with respect to the cameras. The end result is a detailed map describing the time-locked trajectories plotted however on a 3D time independent plane.

## Keywords

Tracking tool, computer vision, mobility assessment, people-tracking, trajectory reconstruction

## Introduction

In accessibility studies that aim to achieve inclusive access for all, there is a need to study and characterise certain motion of interest undertaken by pedestrians, wheelchairs users, etc. An example is a study of obstacle avoidance strategies in human walking behaviour using the Vicon motor system to track trajectories made by participants (Huber et al., 2014). Existing motion-tracking systems such as the aforementioned Vicon Motion System for 3D motion tracking and the CODA Motion System for gait analysis can be used to produce the required motion trajectories. However, a typical setup for these motion systems involves very high prohibitive costs per cubic meter of measurement space. Thus conducting researches both in a small and large scale is typically restricted to high-end facilities or research operations. Low cost visual techniques on the other hand, such as blob counting are used to track participants in accessibility studies (Isard & MacCormick, 2001). Here, however, information such as participant orientation is not available and only a few participants can be tracked using such a technique. MoRe-T2 on the other hand can track as well as identify individual markers.

This paper suggests a low-cost portable and modular alternative tracking toolkit called, MoRe-T2 (Mobility Research Tracking Toolkit) that allows for easy tracking and trajectory characterisation taking orientation into account. Its portability stems from the fact that a minimum working system only needs cameras and a computer. A modular design allows most cameras to be used either via USB or a network setup. The system uses the ARToolkitPlus library (Daniel Wagner, 2007), which outputs pose information of unique QR code-like patterns, to identify and track these patterns' poses observable from post-processing time-synchronized video recordings of the motion of interest. The ARToolkitPlus on its own is not capable of generating trajectories from multiple cameras using the same inertial frame of reference. Moreover, it was originally developed for a VGA (640x480 pixels) camera resolution and we have extended and validated its capabilities for use with a 3-megapixel camera (2048x1536 pixels). In addition, ARToolkitPlus produces poses that are inherently noisy, depending on the lighting condition, distance to the camera, quality of camera, etc and these have been accounted for in our tracking toolkit.

Up to 4096 unique QR code-like patterns can be printed on paper, and used as tracking markers fixed on participant/wheelchair's bodies. As long as a marker is visible to at least one camera throughout a motion recording, MoRe-T2 can generate the position and orientation of the marker. In future work, we aim to employ a filtering technique to correct for temporary marker occlusions (Karavasilis, Nikou, & Likas, 2010). The system currently works on Linux, Macintosh, and Windows operating systems.

In the following sections, we give a high level guideline into MoRe-T2 setup procedure. This is followed by a comparison and validation of its performance and characteristics such as accuracy, against those of the CODA Motion system. We then provide a specific case study of using MoRe-T2 to track wheelchair motion.

**Method**

In this section, we detail the steps taken to setup and produce trajectory results with the MoRe-T2, which includes principles in camera selection and calibration. MoRe-T2 only requires the below hardware for a minimum working system:

1. Camera (1+): For recording motion.
2. Computer: For data processing and system control.

The procedure for using the system essentially entails first mounting the cameras at fixed positions from whence markers can be observed as shown in Figure 1. The markers are attached to the moving participant(s) and a video of the motion is recorded and then processed by the computer. MoRe-T2 outputs absolute time stamps that can be used to synchronize motion with data from other devices such as an inertial sensor.

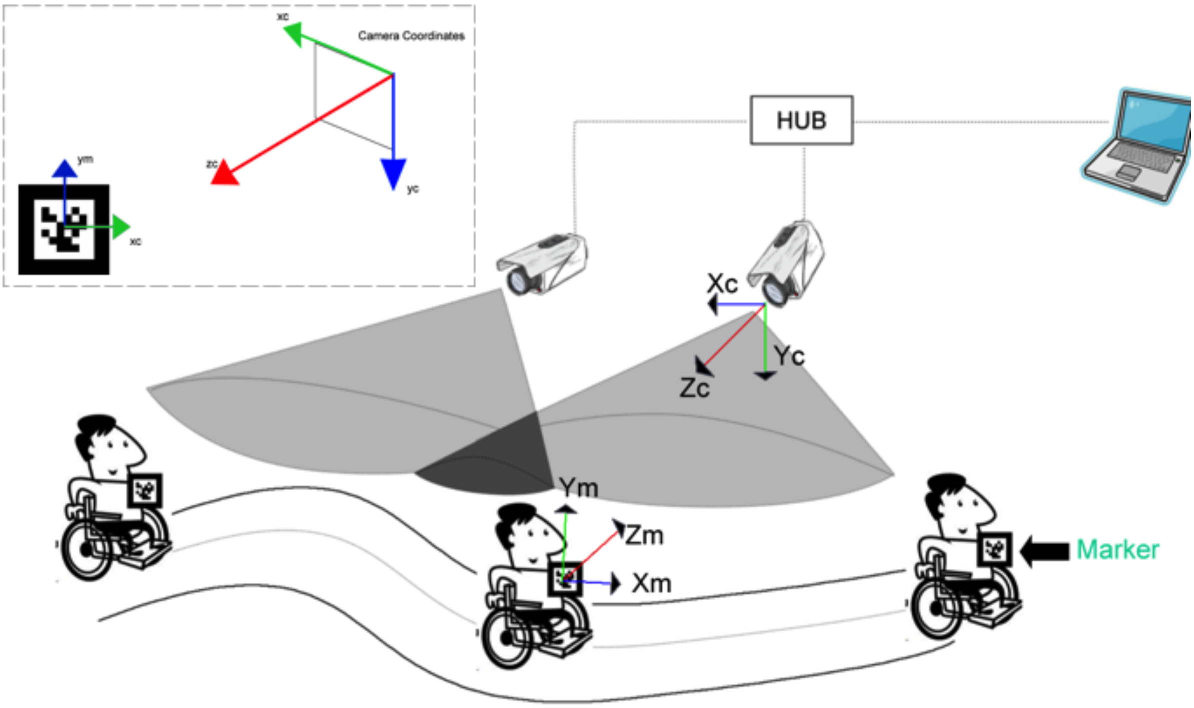



Figure 1 shows the camera and marker axis, and the general setup for the MoRe-T2 system.

To achieve these however, it is very important to carry out the following one-time preliminary procedures in the order listed:

Camera Selection

A camera is selected based on qualities, which can be separated into connectivity, optics, and software compatibility. The suitable qualities are summarised in the Appendix A. For example, shorter exposure time may be preferable to reduce blur in motion capture (Raskar, Agrawal, & Tumblin, 2006).

To use these qualities in selecting a camera, we can employ a mathematical rule of thumb (Peterson, 2009) that gives the maximum distance from a camera for a combination of camera characteristics and marker size as shown in the case study section. This is very useful to ensure that camera selected is capable of working at intended distances.

$$Distance_{max} = \frac{Focal\ Length * Camera\ Resolution * Marker\ Width}{sensor_{size} * 25} \tag{1}$$

Where,

- Distance is the maximum desired measurement distance in mm.
- Camera resolution used here should be the minimum side measurement of the camera sensor's resolution in pixels, which is different from the lens resolution.
- Sensor size is the camera's sensor size.
- The number 25 represents the minimum detectable resolution of a marker in pixels (Kohler, Pagani, & Stricker, 2011).
- Marker width, focal length, sensor size and measurement distance are all in mm.

Camera Distortion Correction/Calibration

For accurate estimation, MoRe-T2 requires that a camera's inevitable curvature and/or distortion be accounted for. To do this, several images of a checkerboard pattern of known size are taken from several angles and positions. A calibration programme, in this case GML C++ Calibration Toolkit (Vezhnevets, Velizhev, Chetverikov, & Yakubenko, 2005), can then use these images to calculate correction parameters, which are very vital to the accuracy of the system. This toolkit outputs an estimate of the camera's focal length, principal axis and distortion parameters, along with a measure of uncertainty in these values as shown in Figure 2.

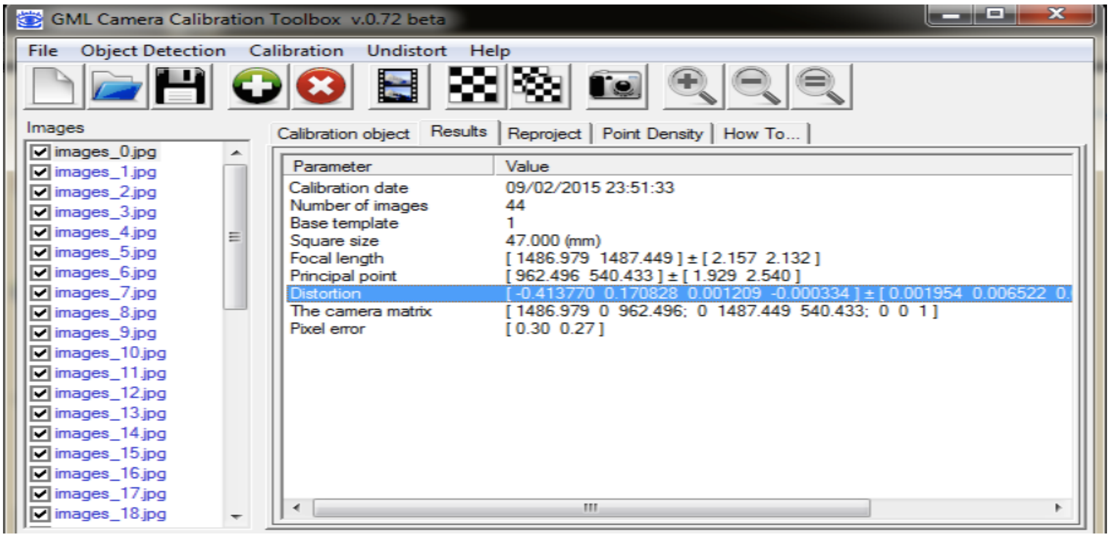



Figure 2 showing a typical camera calibration result

MoRe-T2 Graphics Interface (GUI)

The GUI shown in Figure 3 provides an easy way to use the tracking system. For IP (internet protocol) cameras, we assign a unique number to a camera and in a dropdown list (not shown), we type in the full IP address of the camera in a format,

[rtsp|http]://<username>:<password>@<Cam_IP>/live.sdp

Where username and password are the login credentials needed to access the IP camera video stream. Cam_IP is the IP address of the camera. A camera's video stream can be viewed in the blue viewing area.

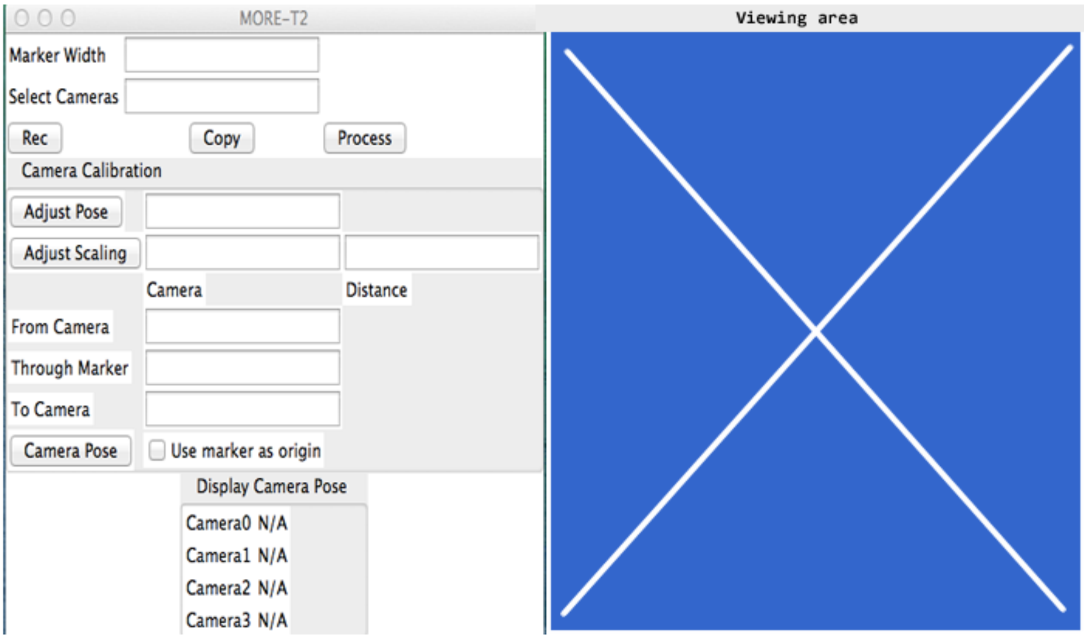



Figure 3 MoRe-T2 Graphical User Interface with a viewing area to observe camera feeds

Camera Adjustment

Critical to the accuracy of the trajectory tracking, all cameras are placed so that their fields of view are all approximately parallel to the major plane of motion. The major plane can for example be a ground surface across which a pedestrian's walking trajectory is recorded. To do this, we place a marker flat on the major plane and adjust the orientation of the camera until both the attitude and heading angles are close to zero (or 360 degrees) as in Figure 4. The MoRe-T2 GUI can show the pose of the camera with respect to the marker, which guides the camera orientation adjustment process. We simply type in the number of the camera into the textbox right of the "Adjust Pose" button and then click the button.

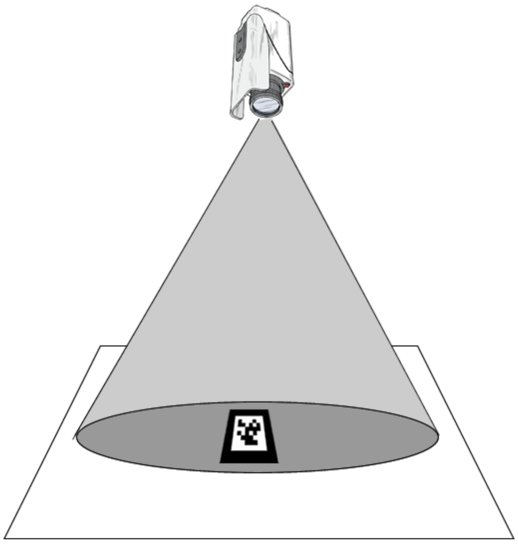


Figure 4 Camera is parallel to the plane of motion.

Scaling Adjustment

The camera distortion calibration although it corrects for lens curvature, may need to be fine-tuned to give a more accurate correspondence between distances in images appearing on the camera screen and the corresponding real world distance. This is because the ARToolkitPlus library used was intended for small distances of a few centimetres and in our case where we require measurements up to a few meters, discrepancies between the camera image size estimates and the actual sizes become significant. Possible causes of this discrepancy include inaccurately measured checkerboard size and marker size. As a result, we need to get a scaling constant that compensates for the discrepancy. To do this, we place two markers on the platform at a fixed distance, a few meters apart while ensuring both markers are visible to the camera as in Figure 5. A marker's position is measured from its centre. We input the real distance and the camera number separated into the GUI and it computes the scaling correction for that camera. This is done for all cameras.

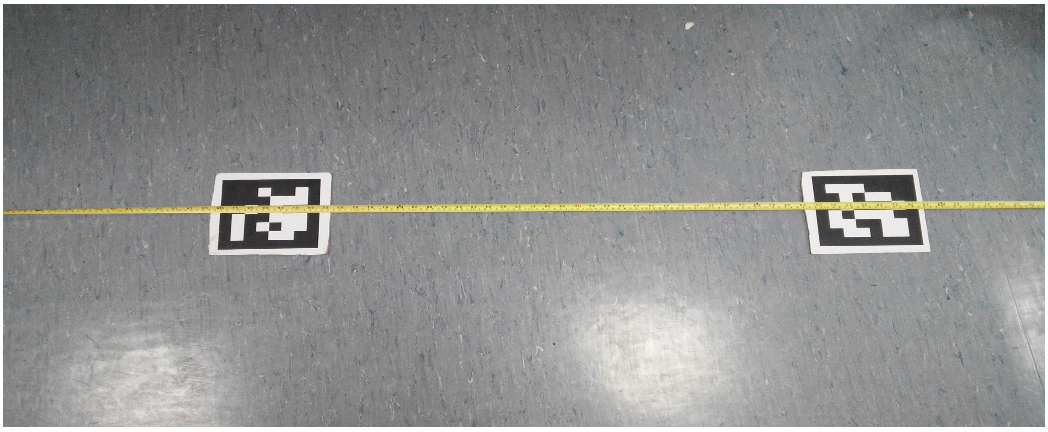


Figure 5 shows two markers, whose distance apart is being measured to obtain a correct scaling factor between the distance estimated by the camera and the real world distance

Camera Pose Estimation

After adjusting the camera and correcting for the scaling factor, we estimate the actual pose of the camera in our chosen inertial frame so that we can. To do this, we first specify measurements' the origin, which can either be that of a specific camera or a marker's axis. If we want a marker as the origin, we simply place the marker as desired and use the GUI to get the pose of the camera. Otherwise, we can pick a camera to serve as the origin's axis. Shown in **Error! Reference source not found.**, obtaining the pose of other cameras is done by placing a marker on the platform at a region that overlaps between two cameras where one's pose is known and the other unknown. The GUI can then be used to get the pose of latter from the former. This process is repeated till all the cameras' poses are known.

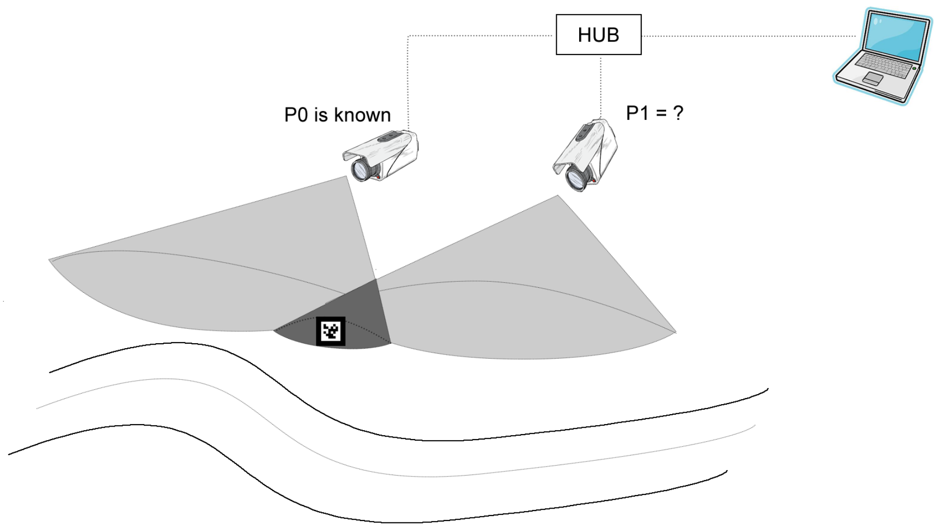



Figure 6 shows camera field of view overlap for pose estimation of an unknown camera from a known camera's position

**Implementations and Results**

In this section, we show two case studies of using MoRe-T2. One in which the MoRe-T2 is validated and compared with the CODA tracking system and another where the tracking system is used to measure the trajectory of the wheelchair using four IP cameras.

Comparison with CODA and Limitations

Important to any tracking system are accuracy, precision, and maximum sampling time, occlusion of marker and ease of use. For the MoRe-T2, these characteristics are very much dependent on the choice of camera and the quality of the camera distortion calibration. In this section, we highlight how the choice of camera and calibration affects desired performance for MoRe-T2 and we compare this with the performance of the CODA system across criteria such as accuracy, sampling time, etc.

Accuracy: this measures how close the measured pose (position and orientation) is from the real world pose. In this regard, the better the calibration of the camera, the better the accuracy.

Lighting conditions also affect the accuracy. A typical well-lit environment, without the paper marker being overexposed or underexposed to light is ideal. However, MoRe-T2 has some robustness to lighting conditions by making automatic adjustments in image contrast.

Resolution of the camera and the marker size also play a role in accuracy. In general, the closer the marker is to the camera or the higher the resolution of the camera is, the higher the measurement's accuracy.

In general, we find that measurements follow a normal distribution about a mean that depends most on the calibration of the camera and a variance that depends most on the lighting condition, and distance from marker. The CODA on the other hand is more much accurate with maximum position error less than two millimetre.

Sampling time: Sampling time depends very much on the frame rate of cameras used as well as the capacity to acquire data from the cameras also known as the system's bandwidth. For a USB camera, the sampling limit is the camera's frame rate, which typically has an upper limit of 30Hz. The limit for an IP camera is more complicated and the analysis though possible, is beyond the scope of this paper. For the CODA, the sampling of the CODA is variable as well but with much higher sampling rate of 800Hz using 6 markers.

Marker Occlusion and Artefacts: All motion-tracking systems track some form of marker. The paper markers of the MoRe-T2 are larger than the markers of the CODA. This means that a MoRe-T2 marker may exhibit more motion artefacts as its area experiences more drag by air current. Moreover, there is a greater chance of placing a marker around moving body parts that can impact on the marker.
In general, the MoRe-T2 markers are suitable for low speed applications whereas the CODA markers can be securely fixed without interference from other moving parts.

Ease of use: For an initial use, the CODA needs the axis of the inertial frame defined, which can be a straightforward process. Three markers are placed orthogonal to each other and any two can be used to define the direction of an axis. Only two axes need to be defined as the third is then inferred using the right hand rule. Any marker can serve as the origin. Compared with the setup procedure of the CODA, the MoRe-T2 requires a bit more effort as shown in the methodology section. This is a trade-off on the portability and low cost of the system.

In terms of user interface, whilst the CODA has more functionality and hence a more sophisticated and complex user interface as shown in Figure **7**. The MoRe-T2 presents a simple interface and functionality for easy motion captures.

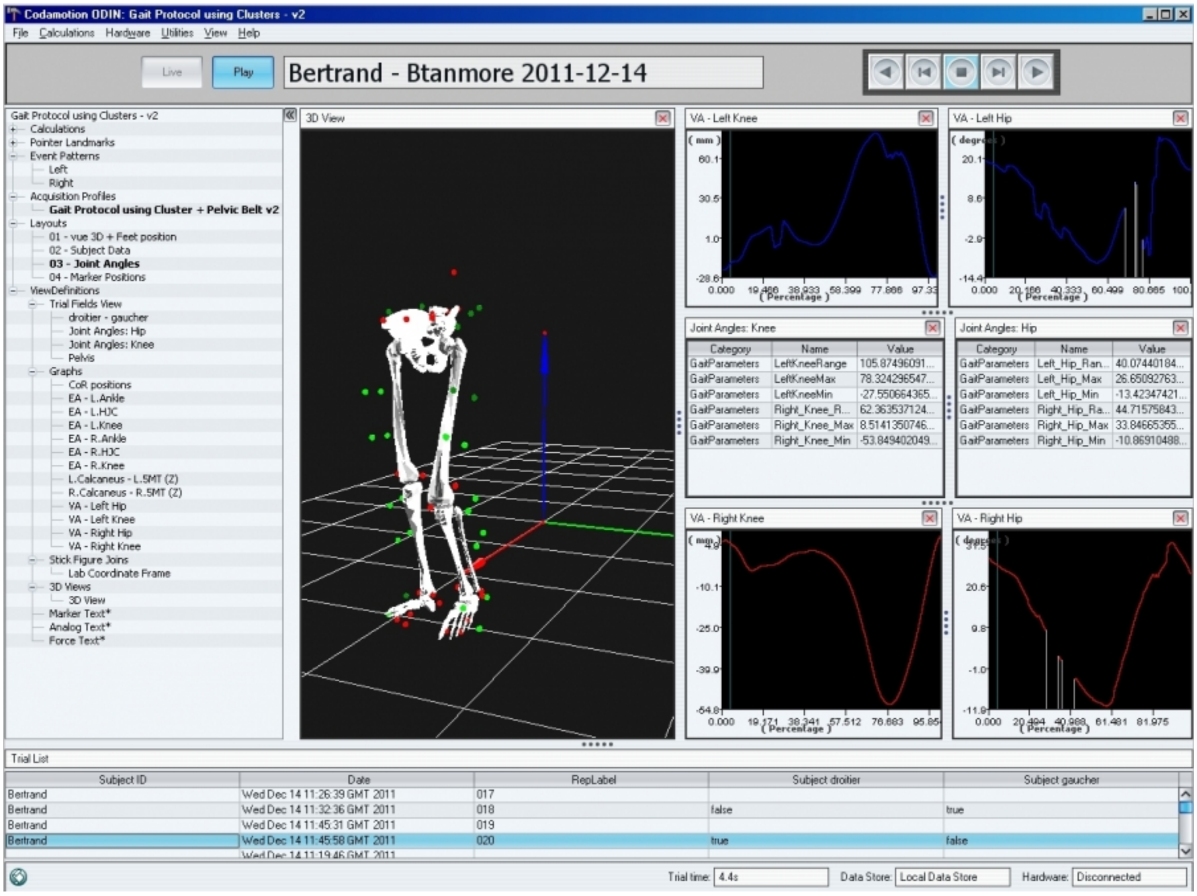



Figure 7 shows the CODA motion user interface, which is more sophisticated and more complex than the MoRe-T2

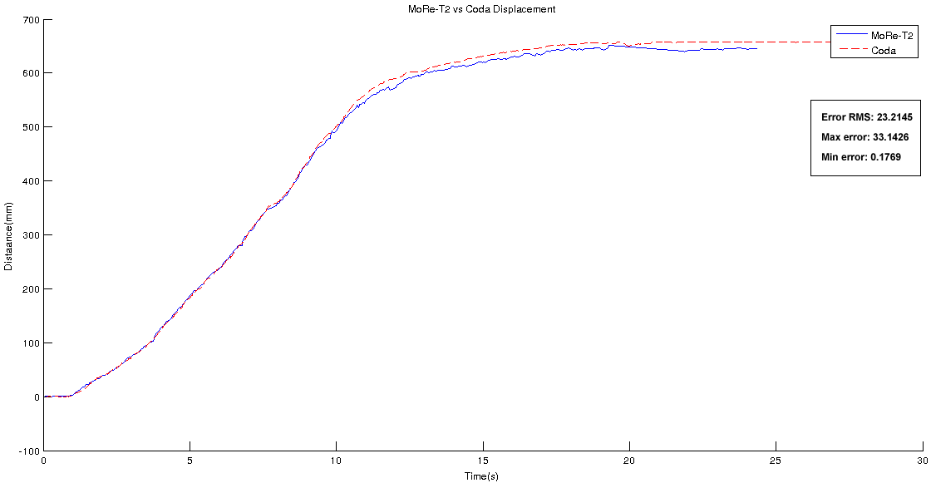
Validation using the CODA

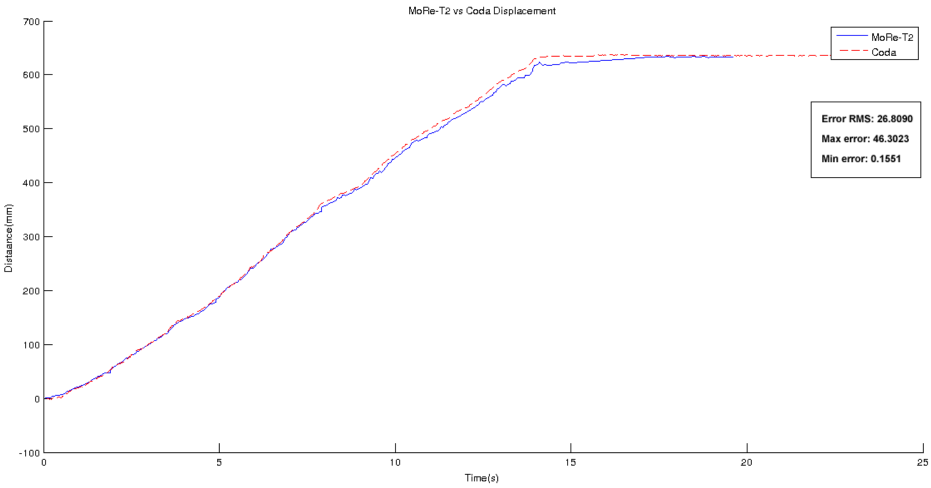
To validate the MoRe-T2, we compare its accuracy with the well-reported performance for the CODA, taken as our ground truth (Moore, 2005). To do this, we attached a CODA marker to a MoRe-T2 maker, performed three repetitions of moving the pair by some distance in a straight line whilst recording the absolute displacement from the start positions. This allowed us to track in real-time any discrepancy in accuracy between the MoRe-T2 and the CODA. Following from Pythagoras theorem, we use the error of absolute displacement $Error_{absolute}$ as the lower bound or worst-case error in the x-axis, $Error_x$ and in the y-axis, $Error_y$ since the z-axis was more or less kept constant (by following the preliminary setup). The formula for the inequality is given below:

$$Error_x = X_{actual} - X_{measured} \tag{2}$$

$$Error_x, Error_y \leq Error_{absolute} = \sqrt{Error_x^2 + Error_y^2} \tag{3}$$

The result of our validation is shown in Figure **8** where we obtained an error root mean squared (RMS) of 38mm. Although measurements are inherently dependent upon choice of camera and calibration, we obtained our result evaluating at 5m from the camera for a marker size of 170x170cm using a 3MP camera. Our lens distortion calibration output gave parameters with accuracy less than 3 pixels.



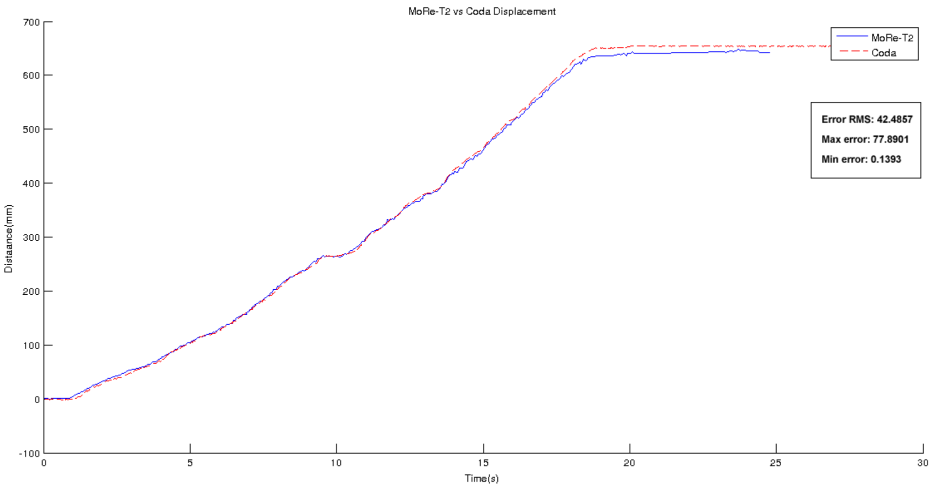



Figure 8 showing the real-time displacement of a coda marker fixed to a MoRe-T2 marker. We can see a discrepancy between the measurement of the Coda (taken as our ground truth) and the More-T2. The maximum error recorded was 78mm while the maximum RMS error for a single movement was 42mm.

Wheelchair Motion Tracking

The aim of this section is to give a concrete example of the potential uses of the MoRe-T2 as relevant in transportation and accessibility studies. In this test scenario, we are interested in simply obtaining the trajectory of a wheelchair moving in an empty corridor in a U-shape. The corridor had natural lighting. Four cameras are fixed to the ceiling; in a U-shape overlooking the intended corridor path and their field of view overlap as shown in Figure **9**. Each camera views an area of 3mx2m. Our maximum measurement distance from the camera is 3m and our marker size is fixed at 110mmx110mm. These two requirements can be respectively fixed by a limitation in room size and the size of the available areas to attach the marker. These requirements are not strict but it is useful to define upper bound in requirements so as to guide the camera selection process.

Figure 9 showing four cameras connected in a U-shape

We chose to use IP cameras for their flexible setup capability and scalability. Alternatively using USB camera can complicate the setup due to the use of repeaters to connect cameras at long distances and the use of a USB hub to gain additional ports for a system with limited USB ports. Thus, we generate a list in Appendix B of possible low cost IP cameras using the rule of thumb.

From Appendix B, Samsung 600 TVL is the cheapest that fits our requirement. However, its shutter speed of 1/50 will produce blurs for moving images, as it is too slow. This leaves the next cheapest option of TRENDnet TV IP310PI.

The TRENDnet cameras are placed in fixed locations to guarantee minimum occlusion of the markers and the preliminary setup procedure is carried out.

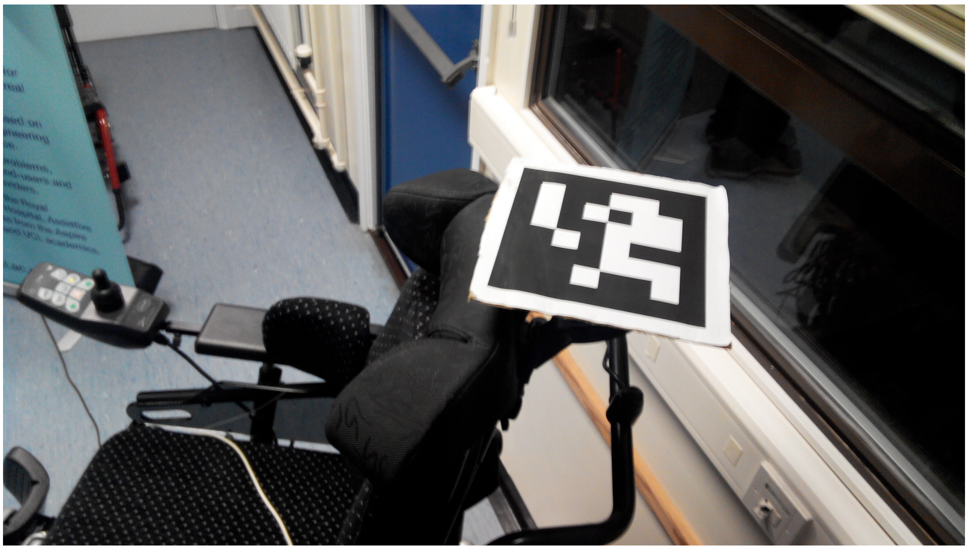


Figure 10 show a marker attached to the top of a wheelchair to track the wheelchairs motion.

A paper printed QR code-like pattern can be glued to cardboard to form a solid flat marker. This marker is then attached to a point on the wheelchair as shown in Figure 10. As origin, another marker is placed in the environment and its axis is used as the inertial frame for our trajectory measurement. Alternatively, we could have used any camera’s axis as the reference frame.

*Result*

Figure 11 and Figure 12 both show the U-shaped motion of the wheelchair. The poses (units are in mm) from each camera have a unique colour code for the marker’s axis. Poses from different cameras overlap nicely and we show 6 of these at the 3 unique regions of camera view overlap. The maximum overlap error is 8cm in the x and y camera axis and about 20cm in the camera z-direction. In **Figure 13**, the wheelchair’s motion direction is shown.

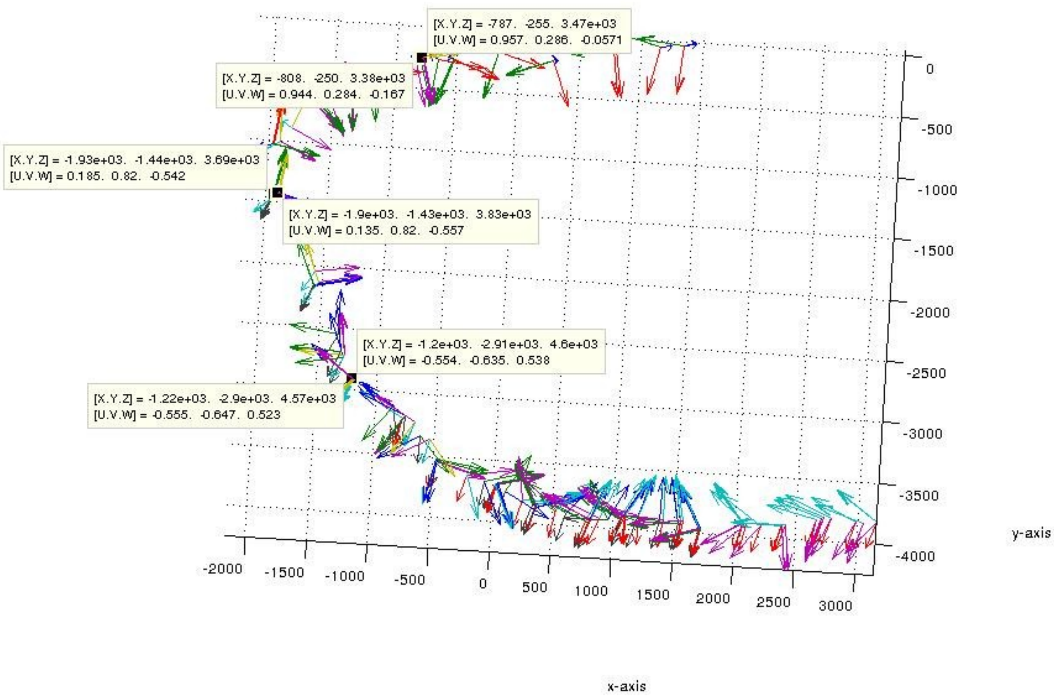


Figure 11 shows top view of wheelchair trajectory. Each camera's field of view overlaps with those of its nearest cameras. There are three regions of overlap and for each overlap region, the figure shows two poses of a single marker produced by the overlapping camera at about the same.

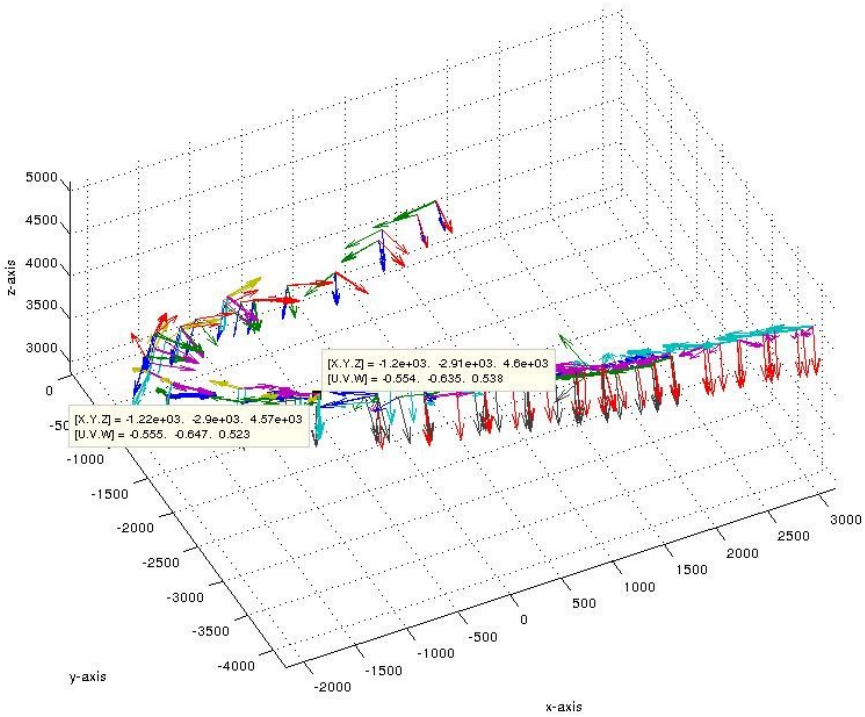



Figure 12 shows a marker pose estimated by two cameras at a region of camera overlap.

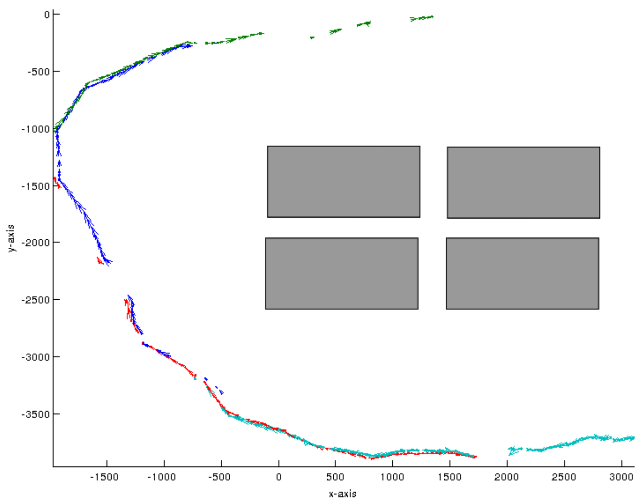



Figure 13 Shows the top view of the corridor. The rectangles are tables that the wheelchair avoid whilst moving.

## Conclusion

We have presented and validated a low cost modular and portable tracking toolkit called Mobile Research Tracking Toolkit (MoRe-T2) based on the ARToolkitPlus library. Findings on measurement for MoRe-T2's accuracy show better performance for closer distances to the camera, higher camera resolution and daylight lighting condition. However, performance is also heavily dependent on calibration, lighting condition and marker size. MoRe-T2 was able to achieve a worst-case peak error of 78mm and an rms error of 42mm for our validation scenario.

## Further Work

We are currently looking to integrate the probabilistic effects of uncertainty to due factors such as camera sensor noise, lighting condition and occlusion. To do this, we can employ a Kalman or particle filter to enable an even more robot tracking capability.

For trajectories continuation amongst multiple cameras, time synchronisation is key. However, the accuracy of the time synchronisation currently depends heavily on the cameras capabilities especially for IP cameras. Some IP camera can synchronise only to the nearest 5 seconds. There is thus a need to develop new techniques to accurately correct for timing offset amongst cameras so that trajectories can be more reliably stitched together enabling time derivatives such as velocity and acceleration to be accurately inferred from motion data.

## References


Chia-Kai, L., Li-Wen, C., & Chen, H. H. (2008). Analysis and Compensation of Rolling Shutter Effect. *Image Processing, IEEE Transactions on, 17*(8), 1323-1330. doi: 10.1109/TIP.2008.925384

Daniel Wagner, D. S. (2007). ARToolKitPlus for Pose Tracking on Mobile Devices. *12th Computer Vision Winter Workshop (CVWW'07*.

Huber, M., Su, Y.-H., Krüger, M., Faschian, K., Glasauer, S., & Hermsdörfer, J. (2014). Adjustments of Speed and Path when Avoiding Collisions with Another Pedestrian. *PLoS ONE, 9*(2), e89589. doi: 10.1371/journal.pone.0089589

Isard, M., & MacCormick, J. (2001, 2001). *BraMBLe: a Bayesian multiple-blob tracker.* Paper presented at the Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on.

Karavasilis, V., Nikou, C., & Likas, A. (2010). Visual Tracking by Adaptive Kalman Filtering and Mean Shift. In S. Konstantopoulos, S. Perantonis, V. Karkaletsis, C. Spyropoulos & G. Vouros (Eds.), *Artificial Intelligence: Theories, Models and Applications* (Vol. 6040, pp. 153-162): Springer Berlin Heidelberg.

Kohler, J., Pagani, A., & Stricker, D. (2011). Detection and identification techniques for markers used in computer vision. *Visualization of Large and Unstructured Data Sets-Applications in Geospatial Planning, Modeling and Engineering, 19*, 36-44.

Moore, A. (2005). The human movement laboratory. doi: http://about.brighton.ac.uk/sohp/research/resources/coda.php?PageId=710

Peterson, M. (2009). How to calculate image resolution.

Raskar, R., Agrawal, A., & Tumblin, J. (2006). Coded exposure photography: motion deblurring using fluttered shutter. *ACM Trans. Graph., 25*(3), 795-804. doi: 10.1145/1141911.1141957

Soldan, S. (2012). On extended depth of field to improve the quality of automated thermographic measurements in unknown environments. *Quantitative InfraRed Thermography Journal, 9*(2), 135-150. doi: citeulike-article-id:11957285
doi: 10.1080/17686733.2012.738110
Tourapis, H. Y. C., & Tourapis, A. M. (2003, 6-9 July 2003). *Fast motion estimation within the H.264 codec.* Paper presented at the Multimedia and Expo, 2003. ICME '03. Proceedings. 2003 International Conference on.
Vezhnevets, V., Velizhev, A., Chetverikov, N., & Yakubenko, A. (2005). GML C++ camera calibration toolbox.

## Appendix

Appendix A

Table 1 shows several criteria used to select a suitable camera for MoRe-T2

| Category | Quality | Details |
|---|---|---|
| **Connectivity** | Data interface | USB cameras are limited to the number of USB ports available on the computer. Alternatively, a USB Hub can be used but here, the number of cameras is limited by bandwidth of the port and power requirements.<br><br>More than to 8 IP cameras can be connected to one computer making it suitable for larger areas. Power over Ethernet (PoE) is especially attractive for IP cameras as it reduces the number of wires needed, encouraging a more flexible and longer distance setup. IP cameras data cables are typically much longer than USB unless repeaters are used for USB connection. |
| **Optics** | Focal Length | A longer focal length means a higher depth region where the image is in focus. However, the field of view becomes narrower (Soldan, 2012). |
| | Lens exposure time | The smaller the exposure time, the less blurred a moving image will be. However, images become darker (Raskar et al., 2006). |
| | Sensor resolution | The higher the resolution of the sensor, the greater the distance from which images can be obtained. |
| | Sensor type | CMOS camera sensor can either be progressive scan, global shutter or rolling scan. Only rolling scan is not ideal since it requires lines of sensors acquiring image data are different times. This means that one picture of a moving image might appear malformed (Chia-Kai, Li-Wen, & Chen, 2008). |
| **Software Compliance** | Compression type | H264 and/or MJPEG compatible devices are most ideal for encoding objects in motion (Tourapis & |

| | | |
|---|---|---|
| | | Tourapis, 2003). |
| | Software Driver | USB camera need to have drivers compatible with the operating system should be used. For IP cameras, most only need a compatible browser to change settings. |

Appendix B
Table 2 displays several example candidate ip cameras that may be selected for an experiment requiring a minimum measurement distance of 3m and a maximum marker size of 110mm x110mm

| Camera | Sensor size | Sensor Type | Focal Length (mm) | Resolution | Shutter speed | Streaming | Max distance for 110mm marker | Price w/ vat |
|---|---|---|---|---|---|---|---|---|
| TRENDnet TV IP310PI Outdoor 3 MP PoE Day/Night Network Camera - network CCTV camera | 1/3” | N/A | 4 | 2048 x 1536 | 1/100,000 | Yes | 3.1928 | £149.99 |
| D-Link DCS 7010L HD Mini Bullet Outdoor Network Camera - network camera | ¼” | Progressive scanning | 4.3 | 1280 x 720 | Variable (up to 1/2000) | Yes | 3.7220 | £199.19 |
| D-Link DCS 7000L Wireless AC Day/Night HD Mini Bullet Cloud Camera - network CCTV camera | 1/4" | Progressive scanning | 2.4 | 1280 x 720 | Variable (up to 1/2000) | | 2.0774 | £137.99 |
| Link DCS-3710 HD WDR PoE Day/Night IP Camera | 1/3" | Progressive scanning | 12 mm | 1280 x 960 | Variable (up to 1/2000) | Yes | 5.9866 | £226.79 |
| TRENDnet TV IP302PI | 1/4" | N/A | 4.2 | 1280 x 720 | Variable | | 2.0954 | £176.39 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Outdoor Megapixel PoE Day/Night Internet Camera - network camera | | | | | (up to 1/2000) | | | |
| TRENDnet TV IP522P ProView MegaPixel PoE Internet Camera - network CCTV camera | 1/4" | N/A | 4 | 1280 x 960 | Variable (up to 1/2000) | Yes | 1.9955 | £129.59 |
| Samsung 600 TVL 2.8-10mm varifocal lens IP66 bullet camera | 1/3" | N/A | 10 | 752 x 582 | 1/50 | No | 3.0243 | £129.59 |

Appendix C
Although metrics are inherently dependent upon choice of camera and calibration, the table below is a typical value we obtained evaluated at 3m and 5m from a camera for a marker size of 170x170cm using a 3MP camera. Our lens distortion calibration output gave parameters with accuracy less than 3 pixels.
Table 3 showing the comparison of performance characteristics of MoRe-T2 and CODA motion system

| **Characteristic** | **MoRe-T2** | **CODA** |
|---|---|---|
| **Resolution** | | |
| Standard deviation of position static marker | At 3m<br>0.1822 in X axis<br>0.0927 in Y axis<br>2.8538 in Z axis<br>0.310 in Attitude<br>0.184 in Heading<br>0.074 in Bank<br><br>At 5m<br>0.5911 in X axis<br>0.2094 in Y axis<br>8.9809 in Z axis<br>3.90283 in Attitude<br>5.2774 in Heading<br>0.2513 in Bank | 0.05mm in X and Z axes<br><br>0.03mm in Y axis |
| **Accuracy** | | |
| Peak to peak deviations from actual | ±12mm in X and Y axis, when stationary | ± 1.5mm in X and Z axes |

| | | |
|---|---|---|
| positions | ±39mm in X and Y axis, when moving<br><br>±38mm in Z-axis when stationary | ± 2.5mm in Y axis |
| Sampling rates: | Camera and interface setup dependent. Typically < 30Hz | 56 sensors - 100Hz<br>28 sensors - 200Hz<br>12 sensors - 400Hz<br>6 sensors - 800Hz |
| Marker Occlusion | Greater likelihood because of larger marker size | Lower likelihood due to small form size |
| Motion Artefacts | Larger surface area incurs more drag force. Use for low speed applications. | Depends on how sturdy the marker is attached. Can be used for high-speed applications. |
| Ease of use | User interface is simple and camera position calibration is easier to perform | User interface is more complex and camera position calibration is more demanding to perform |