

# MEASURING COMPLETE GROUND-TRUTH DATA AND ERROR ESTIMATES FOR REAL VIDEO SEQUENCES, FOR PERFORMANCE EVALUATION OF TRACKING, CAMERA POSE AND MOTION ESTIMATION ALGORITHMS

R. Stolkin<sup>a\*</sup>, A. Greig<sup>b</sup>, J. Gilby<sup>c</sup>

<sup>a</sup> Center for Maritime Systems, Stevens Institute of Technology, Hoboken, NJ 07030 USA – RStolkin@stevens.edu

<sup>b</sup> Dept. of Mechanical Engineering, University College London, WC1E 6BT UK – a\_greig@meng.ucl.ac.uk

<sup>c</sup> Sira Ltd., South Hill, Kent BR7 5EH, UK – john.gilby@sira.co.uk

**KEY WORDS:** Vision, robotics, tracking, navigation, registration, calibration, performance, accuracy.

## ABSTRACT:

Fundamental tasks in computer vision include determining the position, orientation and trajectory of a moving camera relative to an observed object or scene. Many such visual tracking algorithms have been proposed in the computer vision, artificial intelligence and robotics literature over the past 30 years. Predominantly, these remain un-validated since the ground-truth camera positions and orientations at each frame in a video sequence are not available for comparison with the outputs of the proposed vision systems. A method is presented for generating real visual test data with complete underlying ground-truth. The method enables the production of long video sequences, filmed along complicated six degree of freedom trajectories, featuring a variety of objects, in a variety of different visibility conditions, for which complete ground-truth data is known including the camera position and orientation at every image frame, intrinsic camera calibration data, a lens distortion model and models of the viewed objects. We also present a means of estimating the errors in the ground-truth data and plot these errors for various experiments with synthetic data. Real video sequences and associated ground-truth data will be made available to the public as part of a web based library of data sets.

## 1. INTRODUCTION

An important and prolific area of computer vision research is the development of visual tracking and pose estimation algorithms. Typically these fit a model to features extracted from an observed image of an object to recover camera pose, track the position and orientation of a moving camera relative to an observed object or track the trajectory of a moving object relative to a camera.

Clearly, proper validation of such algorithms necessitates test images and video sequences with known ground-truth data, including camera positions and orientations relative to the observed scene at each frame, which can be compared to the outputs of proposed algorithms in order to compute errors. Surprisingly, very few such data sets or methodologies for creating them are discussed in the literature, with reported vision systems often validated in ad hoc ways.

Many papers attempt to demonstrate the accuracy of tracking algorithms by superimposing, over the observed image, a projection of the tracked object based on the positions and orientations output by the algorithm. In fact it can be shown (Stolkin 2004) that even very close 2D visual matches of this kind can result from significantly erroneous 3D tracked positions. One reason for this is that certain combinations of small rotations and translations, either of cameras or observed objects in 3D space, often make little difference to the resulting 2D images. This is especially true for objects with limited features and simple geometry. Such errors can only be properly identified and quantified by means of test images with accompanying complete 3D ground-truth.

It is relatively simple to construct artificial image sequences, with pre-programmed ground-truth, using commonly available

graphics software (e.g. POV-Ray for windows) and this is also common in the literature. However, although testing computer vision algorithms on synthetic scenes allows comparison of performance, it gives only a limited idea of how the algorithms will perform on real scenes. Real cameras and real visibility conditions result in many kinds of noise and image degradation, far more complicated than Gaussian noise or “salt and pepper” speckling and it is not trivial or obvious how to realistically synthesise real world noise in an artificial image (Rokita, 1997; Kaneda, 1991). This becomes even more difficult when the scene is not viewed through clear air but through mist, smoke or turbid water. Artificial scenes do not completely reproduce the detailed variation of objects, the multitude of complex lighting conditions and modes of image degradation encountered in the real world. Vision and image processing algorithms often seem to perform much better on artificial (or artificially degraded) images than on real images. The only true test of computer vision algorithms remains their performance on real data.

To this end, several researchers have attempted to combine real image data with some knowledge of ground-truth. Otte, 1994, describes the use of a robot arm to translate a camera at known speeds, generating real image sequences for the assessment of optical flow algorithms. The measured ground-truth data is limited to known optic flow fields rather than explicit camera positions and the camera is only translated. Rotational camera motion is not addressed. McCane, 2001, also describes image sequences with known ground-truth motion fields. The work is limited to simple 2D scenes containing planar polyhedral objects against a flat background. The technique involves laborious hand-labelling of features in each image and so only very short sequences are usable. Wunsch, 1996, uses a robot arm to position a camera in known poses relative to an observed object. Similarly, Sim, 1999, generates individual images from known camera positions using a camera mounted on a gantry

robot. In the work of both Wunsch and Sim, ground-truth positions are only measured for individual still images as opposed to video sequences. Both authors appear to obtain camera positions from the robot controller. It is not clear if or how the positions of the camera (optical centre) were measured relative to the robot end-effector. Agapito, 2001, generates ground-truth image sequences using their “Yorick” stereo head/eye platform. The work is limited to providing rotational motion with only two degrees of freedom. Although data for angles of elevation and pan can be extracted from the motor encoders of the platform, these are not in relationship to a particular observed object. The translational position of the camera remains unknown. Maimone, 1996, discusses various approaches for quantifying the performance of stereo vision algorithms, including the use of both synthetic images and real images with various kinds of known ground-truth. Maimone does mention the use of an image of a calibration target to derive ground-truth for a corresponding image of a visually interesting scene, filmed from an identical camera position. However, the techniques are limited to the acquisition of individual, still images from fixed camera positions. The additional problems, of generating ground-truth for extended video sequences, filmed from a moving camera, are not addressed.

In contrast, our method enables the production of long video sequences, filmed along a six degree of freedom trajectory, featuring a variety of objects, in a variety of different visibility conditions, for which complete ground-truth data is known including the camera position and orientation at every image frame, intrinsic camera calibration data, a lens distortion model and models of the viewed objects.

## 2. METHOD

### 2.1 Apparatus and procedure

An industrial robot arm (six degree of freedom Unimation PUMA 560) is used to move a digital cam-corder (JVC GR-DV2000) along a highly repeatable trajectory. “Test sequences”, (featuring various objects of interest in various different visibility and lighting conditions), and “calibration sequences” (featuring planar calibration targets in good visibility) are filmed along identical trajectories (figures 1, 2).



Figure 1. “Test sequence”-camera views a model oil-rig object in poor visibility.



Figure 2. “Calibration sequence”-camera views calibration targets in good visibility.

A complete camera model, lens distortion model, and camera position and orientation can be extracted from the calibration sequence for every frame, by making use of the relationship between known world co-ordinates and measured image co-ordinates of calibration features. This information is used to provide ground-truth for chronologically corresponding frames in the visually interesting test sequences. Objects to be observed are measured, modeled and located precisely in the co-ordinate system of one of the calibration targets.

For those researchers interested in vision in poor visibility conditions (e.g. Stolkin 2000) dry ice fog can be used during the “test” sequences (figure 1) in addition to various lighting conditions (e.g. fixed lighting or spot-lights mounted on and moving with the camera).

Note, it is not feasible to extract camera positions from the robot control system since the position of the camera relative to the terminal link of the robot remains unknown; industrial robots, while highly *repeatable*, are not *accurate*; chronologically matching a series of robot positions to a series of images may be problematic.

### 2.2 Synchronisation

The “calibration” and “test” sequences are synchronised by beginning each camera motion with a view of an extra “synchronisation spot” feature (a white circular spot on black background). A frame from each sequence is found such that the “synchronisation spot” matches well when the two frames are superimposed. Thus the  $n^{\text{th}}$  frame from the matching frame in the test sequence is taken to have the same camera position as that measured for the  $n^{\text{th}}$  frame from the matching frame in the calibration sequence. The two sequences can only be synchronised to the nearest image frame (i.e. a worst case error of  $\pm 0.02$  seconds at 25 frames per second). There are two ways of minimizing this error. Firstly, the camera is moved slowly so that temporal errors result in very small spatial errors. Secondly, many examples of each sequence are filmed, increasing the probability of finding a pair of sequences that match well (correct to the nearest pixel). If ten examples of each sequence are filmed, then the expected error is reduced by a factor of 100.

### 2.3 Feature extraction and labelling

The calibration targets are black planes containing square grids of white circular spots. The planes are arranged so that at least one is always in view and so that they are not co-planar. The positions of spots in images are determined by detecting the spots as “blobs” and then computing the blob centroid. A small number (at least 4) of spots in each of a few images scattered through the video sequence are then hand-labeled with their corresponding target plane co-ordinates. The remaining spots in all images are labeled by an automated process. The initial four labels are used to estimate the homography mapping between the target plane and the image plane. This homography is then used to project all possible target spots into the image plane. Any detected spots in the image are then assigned the labels of the closest matching projected spots. Spots in chronologically adjacent images are now labeled by assigning them the labels of the nearest spots from the previous (already labeled) image. These two processes, of projection and propagation, are iterated backwards and forwards over the entire image sequence until no new spot labels are found.

### 2.4 Camera calibration and position measurement

Our calibration method is adapted from that of Zhang, 1998, which describes how to calibrate a camera using a few images of a planar calibration target. Related calibration work includes Tsai, 1987. The following is a condensed summary of our implementation of these ideas.

#### 2.4.1 Homography between an image and a calibration target:

Since the calibration targets are planar, the mapping between the (homogeneous) target co-ordinates of calibration features,  $\mathbf{X}_t = [X_t \ Y_t \ 1]^T$ , and their corresponding (homogeneous) image co-ordinates,  $\mathbf{x}_i = [u \ v \ 1]^T$ , must form a homography, expressible as a  $3 \times 3$  matrix:

$$\mathbf{x}_i = \mathbf{H}\mathbf{X}_t = [\mathbf{h}_1 \ \mathbf{h}_2 \ \mathbf{h}_3]\mathbf{X}_t \quad (1)$$

Thus each calibration feature, whose position in an image is known and whose corresponding target co-ordinates have been identified, provides two constraints on the homography. A large number of such feature correspondences provides a large number of simultaneous equations:

$$\begin{bmatrix} w_1 u_1 & w_2 u_2 & \dots & w_n u_n \\ w_1 v_1 & w_2 v_2 & \dots & w_n v_n \\ w_1 & w_2 & \dots & w_n \end{bmatrix} = \mathbf{H} \begin{bmatrix} X_1 & X_2 & \dots & X_n \\ Y_1 & Y_2 & \dots & Y_n \\ 1 & 1 & \dots & 1 \end{bmatrix} \quad (2)$$

A least squares fit homography is then found using singular value decomposition.

#### 2.4.2 Constraints on the camera calibration parameters:

The mapping between the target and image planes must also be defined by the intrinsic and extrinsic camera calibration parameters of the camera:

$$\mathbf{x}_i = \mathbf{H}\mathbf{X}_t = \mathbf{C}\mathbf{E}\mathbf{X}_t \quad (3)$$

where  $\mathbf{C}$  is the “intrinsic” or “calibration matrix”:

$$\mathbf{C} = \begin{bmatrix} f k_u & 0 & u_0 \\ 0 & f k_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

( $f$  is focal length,  $k_u$  and  $k_v$  are pixels per unit length in the  $u$  and  $v$  directions,  $(u_0, v_0)$  are the co-ordinates of the principal point, pixel array assumed to be square) and  $\mathbf{E}$  is the “extrinsics matrix” defining the position and orientation of the camera (relative to the target co-ordinate system), i.e.  $\mathbf{E} = [\mathbf{r}_1 \ \mathbf{r}_2 \ \mathbf{T}]$ , where  $\mathbf{r}$  and  $\mathbf{T}$  denote rotation and translation vectors. Note that only two rotation vectors (not three) are needed since the calibration target plane is defined to lie at  $Z = 0$  in the target co-ordinate system. Hence:

$$\mathbf{H} = [\mathbf{h}_1 \ \mathbf{h}_2 \ \mathbf{h}_3] = \mathbf{C}[\mathbf{r}_1 \ \mathbf{r}_2 \ \mathbf{T}] \quad (4)$$

Since the column vectors of a rotation matrix are always mutually orthonormal, we have:

$$\mathbf{r}_1^T \mathbf{r}_2 = 0 \quad (5)$$

$$\mathbf{r}_1^T \mathbf{r}_1 = \mathbf{r}_2^T \mathbf{r}_2 \quad (6)$$

Since  $\mathbf{r}_n = \mathbf{C}^{-1} \mathbf{h}_n$  these become:

$$\mathbf{h}_1^T \mathbf{C}^{-T} \mathbf{C}^{-1} \mathbf{h}_2 = 0 \quad (7)$$

$$\text{and} \quad \mathbf{h}_1^T \mathbf{C}^{-T} \mathbf{C}^{-1} \mathbf{h}_1 = \mathbf{h}_2^T \mathbf{C}^{-T} \mathbf{C}^{-1} \mathbf{h}_2 \quad (8)$$

Thus one homography provides two constraints on the intrinsic parameters. Ideally, many homographies (from multiple images of calibration targets) are used and a least squares fit solution for the intrinsic parameters is found using singular value decomposition.

Once the intrinsic parameters have been found using a few different views of a calibration target, the extrinsic parameters can be extracted from any other single homography, i.e. the camera position and orientation can be extracted for any single image frame provided that it features several spots from at least one target.

**2.4.3 Locating targets relative to each other:** We use multiple calibration targets to ensure that at least one target is always in view during complicated (six degree-of-freedom) camera trajectories. Provided that at least one target is visible to the camera at each frame, the position of the camera can be computed by choosing one target to hold the world co-ordinate system and knowing the transformations which relate this target to the others. The relationship between any two targets is determined from images which feature both targets together, by determining the homography which maps between the co-ordinate systems of each target. For two targets,  $A$  and  $B$ :

$$\mathbf{x}_i = \mathbf{H}_A \mathbf{X}_A = \mathbf{H}_B \mathbf{X}_B \quad (9)$$

where  $\mathbf{X}_A$  and  $\mathbf{X}_B$  are the positions of a single point in the respective co-ordinate system of each target. Thus:

$$\mathbf{X}_A = (\mathbf{H}_A)^{-1} \mathbf{x}_i = (\mathbf{H}_A)^{-1} \mathbf{H}_B \mathbf{X}_B \quad (10)$$

**2.4.4 Modeling lens distortion:** Lens distortion is modelled as a radial shift of the undistorted pixel location  $(u, v)$  to the distorted pixel location  $(\hat{u}, \hat{v})$ , such that:

$$\hat{u} = u + (u - u_0)(k_1 r^2 + k_2 r^4) \quad (11)$$

$$\text{and} \quad \hat{v} = v + (v - v_0)(k_1 r^2 + k_2 r^4) \quad (12)$$

where  $r^2 = (u - u_0)^2 + (v - v_0)^2$

### 2.4.5 Refining parameter measurements with non-linear optimization:

In practice, all important parameter measurements (camera intrinsics, lens distortion, target to target transformations, camera positions), which are initially extracted using the geometrical and analytical principles outlined above, can be further improved using non-linear optimisation. An error function is minimised, consisting of the sum of the squared distances (in pixels) between the observed image locations of calibration features and the locations predicted given the current estimate of the parameters being refined. This results in a maximum likelihood estimate for all parameters.

Firstly a small set (about 20) of images are used to compute camera intrinsic parameters, lens distortion parameters, camera position and orientation for each image (of the small set) and the transformations between the co-ordinate systems of each target. These parameters are then mutually refined over all views of all targets present in all images of the set, by minimising the following error function:

$$\sum_{\text{target } t=1}^n \sum_{\text{spot } s=1}^m \left\| \mathbf{x}_{\text{image}_{ts}} - \hat{\mathbf{x}}_{\text{image}_{ts}}(\mathbf{C}, k_1, k_2, \mathbf{R}_t, \mathbf{T}_t, \mathbf{X}_{\text{target}_{ts}}) \right\|^2 \quad (13)$$

Where, for  $m$  points (spot centres) extracted from  $n$  target views,  $\mathbf{x}_{\text{image}_{ts}}$  is the observed image in pixelated camera co-ordinates of the world co-ordinate target point  $\mathbf{X}_{\text{target}_{ts}}$ , and  $\hat{\mathbf{x}}_{\text{image}_{ts}}$  is the expected image of that point given the current estimates of the camera parameters  $(\mathbf{C}, k_1, k_2, \mathbf{R}_t, \mathbf{T}_t)$ . Note that the values of the co-ordinates of  $\mathbf{X}_{\text{target}_{ts}}$  are also dependent on the current estimates of target-to-target transformations and these transformations are also being iteratively refined.

Secondly, using the refined values for intrinsics, lens distortion parameters and target-to-target transformations, the camera position and orientation is computed for a single image taken from the middle of the ‘‘calibration sequence’’, again using analytical and geometrical principles. Keeping all other parameters constant, the six-degrees of freedom of this camera location are now non-linearly optimized, minimizing the error between the observed calibration feature locations and those predicted given the current estimate of the camera location and the fixed values (previously refined) of all other parameters.

Lastly, the camera position for the above single image is used as an initial estimate for the camera positions in chronologically adjacent images (previous and subsequent images) in the video sequence. These positions are then themselves optimized, the refined camera positions then being propagated as initial estimates for successive frames, and so on throughout the entire video sequence, resulting in optimized camera positions for every image frame along the entire camera trajectory.

## 3. RESULTS

### 3.1 Constructed data sets

We have filmed video sequences of around 1000 frames (at 25 frames per second) along a complicated six degree-of-freedom camera trajectory. Figure 3 shows the camera position at each frame, as calculated from the calibration sequence. The trajectory is illustrated in relation to the spots of the three calibration targets (30mm spacing between spots).

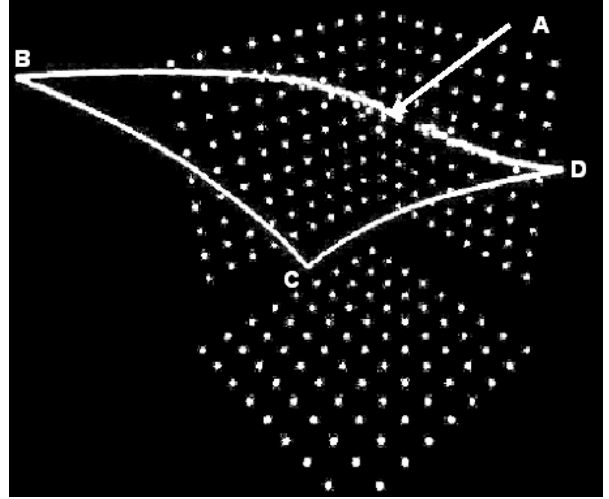


Figure 3. The computed trajectory for a six-degree of freedom of motion video sequence.

The sequences feature various different known (measured and modelled) objects (figure 4) in various different visibility and lighting conditions as well as a corresponding calibration sequence. Analysis of the calibration sequence has yielded a complete camera model, lens distortion model and a camera position and orientation for every frame in each of these sequences.

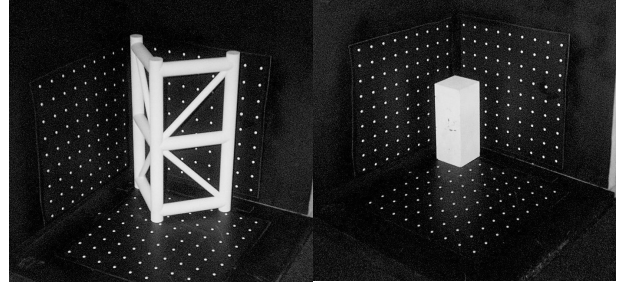


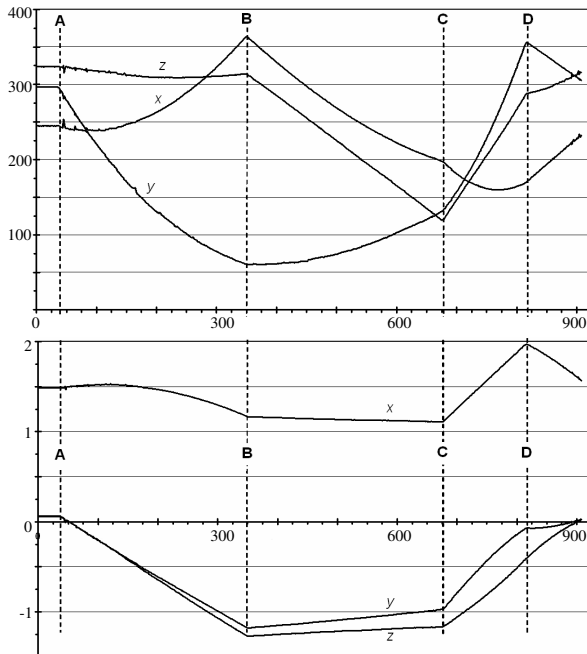
Figure 4. Two of the objects filmed in the video sequences, block and model oil-rig.

### 3.2 Smoothness of trajectory

One indicator of accuracy is the smoothness of the measured trajectory. Figure 3 is a useful visual representation of the trajectory and figures 5 and 6 are plots of the translational and rotational camera co-ordinates at each frame. Points A, B, C, D are corresponding way mark points between figures 3, 5 and 6.

For about the first 40 frames, the camera is stationary at point A. It will be noticed that small sections of the trajectory appear somewhat broken and erratic, approximately frames 40 – 160 and 880 – 910. These ranges correspond to the beginning and end of the trajectory during which the camera is moved from (and back towards) a position fixated on the ‘‘synchronization spot’’ (see section 2.2) at point A. During these periods, comparatively few calibration features are in the field of view. These sections of the video sequence do not correspond to visually interesting portions of the image sequence and are not used for testing vision algorithms. They are included only for synchronization. The remainder of the measured trajectory is extremely smooth, implying a high degree of precision. The robot is old, and its dynamic

performance less than perfect, so the disturbance just after motion is initiated (shortly after point A) is probably due to the inertia of the system. Second and third peaks of decaying magnitude at exactly 20 and 40 frames later suggest that they have a mechanical origin.



Figures 5 & 6. Top graph shows translational components of camera motion along x, y and z axes. Vertical scale in mm. Bottom graph shows rotational components of camera motion about x, y and z axes. Vertical scale in radians. For both graphs, the horizontal scale is image frame number.

### 3.3 Robot repeatability

In order to assess repeatability, the robot was moved along a varied, six-degree of freedom motion that included pauses at three different positions during the motion. Several video sequences were filmed from the robot-mounted camera while moving in this fashion. Images from different sequences, filmed from the same pause positions, were compared. Superimposing the images reveals an error of better than  $\pm$  one pixel. This implies that errors in image repeatability due to robot error approach the scale of the noise associated with the camera itself. Our robot is approximately twenty years old. Modern machines should produce even smaller errors.

### 3.4 Accuracy of scene reconstruction

In order to assess accuracy, the image positions of calibration features were reconstructed by projecting their known world coordinate positions through the measured camera model placed at the measured camera positions. Comparing these predicted image feature positions with those observed in the real calibration sequence yielded an rms error of 0.6 pixels per calibration feature (spot).

When some of the observed objects have been reconstructed in the same way, the errors are worse. Figure 7 shows an image from a sequence featuring a white block object. The measured camera position for the image frame has been used to project a predicted image (shown as a wire frame model) and this

predicted image has been superimposed over the real image. This helps illustrate the errors involved (in this case  $\pm$  3 pixels discrepancy in block edges). This disparity in error magnitude (compared to 0.6 pixels above) may be due to over-fitting of the camera model to features in the calibration target planes and under-fitting to points outside those planes.

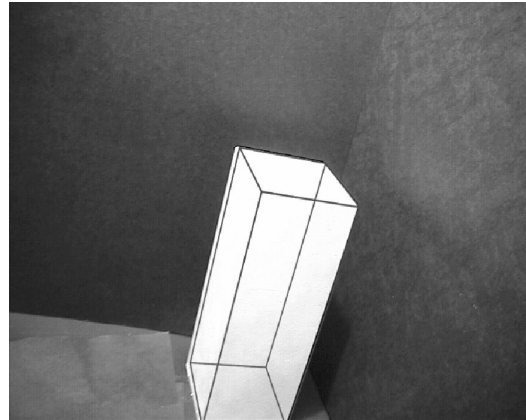


Figure 7. An image from a sequence featuring a block object. The superimposed wire frame image corresponds to the predicted image given the measured camera co-ordinates.

### 3.5 Accuracy of camera pose measurement

In order to estimate the potential overall accuracy of measured camera positions, we have used synthetic calibration data. Although, in general, synthetic images do not reproduce the noise inherent in real images, calibration sequences are filmed in highly controlled conditions which are more reasonably approximated by synthetic images. Graphics software (POV-Ray for windows) was used to generate computer models of calibration targets. A series of synthetic images were then rendered which would correspond to those generated by a camera viewing the targets from various positions. These images were fed into the calibration scheme. Ground-truth as measured by our calibration scheme was then compared with the pre-programmed synthetic ground-truth in order to quantify accuracy. For simplicity, we have used a synthetic camera array of 1000 by 1000 pixels-somewhat better than current typical real digital video resolution but far worse than typical real single image resolution. Over a set of 6 images filmed from several different ranges, but all featuring views of three approximately orthogonal calibration targets (see second paragraph of section 4), the error in measured principal point position was 1.76 pixels and the error in measured focal length was 0.06%. The average error in measured camera position was 1.38mm and 0.024 degrees.

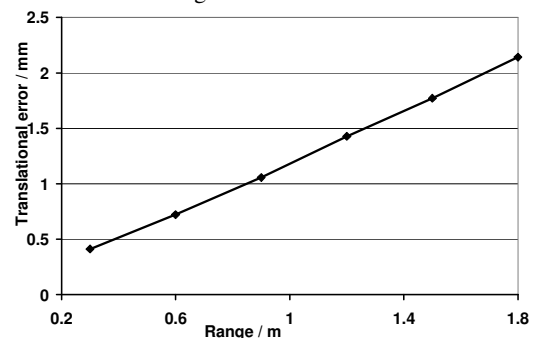
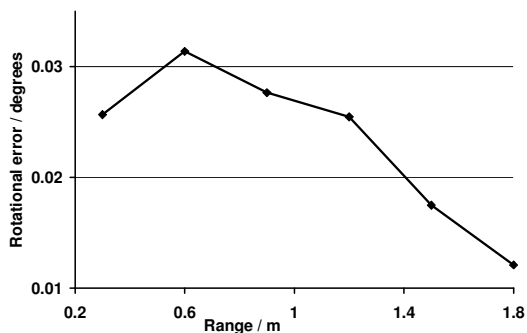


Figure 8. Variation in translational camera position error with range from calibration targets.



Figures 9. Variation in camera orientation error with range from calibration targets.

Figures 8 and 9 plot the variation of error with distance of the camera from the calibration target origin.

#### 4. SUGGESTED IMPROVEMENTS

The problem, outlined in section 3.4, of over-fitting the camera model to points lying in the calibration target planes should be avoided in future work by using calibration images filmed at a variety of different ranges from the calibration targets.

Although it should be possible to determine the position of a calibrated camera given a view of a single calibration target (Zhang, 1998), in practice various small coupled translations and rotations of the camera can result in very similar views, causing measurement uncertainty. These errors can be constrained by ensuring that, throughout the motion of the camera, all three targets, positioned approximately orthogonally to each other, are always in view. In our original experiments with real video sequences, only one or two targets were viewed in most images and so our camera position accuracies are worse than can be achieved. Future researchers should ensure that the camera can always view three, approximately orthogonal, calibration targets in every image.

It is possible to further automate the labeling of calibration spots. By making a specific point, or points, on each target a different colour, it may be possible to eliminate the need to hand-label a small number of spots in each video sequence.

Viewing the "synchronization spot" after the camera has already started moving would eliminate the mechanical vibration problems of the step response noted at the start of the robot's motion.

The synchronisation problem (see section 2.2), that two sequences can only be synchronised to the nearest image frame (i.e. worst case error of  $\pm 0.02$  seconds at 25 frames per second), might be eliminated by triggering the camera externally with a signal from the robot controller such that video sequences started at a specific location in the trajectory.

Note that test sequences can be filmed which feature virtually any kind of object. Even deforming or moving objects could conceivably be used although measuring ground-truth for the shapes and positions of such objects would pose additional challenges. Specifically, the use of objects with known textures might benefit researchers with an interest in surface reconstruction or optic flow. With appropriate equipment, it should also be possible to create real underwater sequences using our technique.

#### 5. CONCLUSION

The field of computer vision sees the frequent publication of many novel algorithms, with comparatively little emphasis placed on their validation and comparison. If vision researchers are to conform to the rigorous standards of measurement, taken for granted in other scientific disciplines, it is important that our community evolve methods by which the performance of our techniques can be systematically evaluated using real data. Our method provides an important tool which enables the accuracy of many proposed vision algorithms, for registration, tracking and navigation, to be explicitly quantified.

#### REFERENCES

- Agapito, L., Hayman, E., Reid, I., 2001. Self-Calibration of Rotating and Zooming Cameras. *International Journal of Computer Vision*. Vol. 45(2), pages 107-127.
- Kaneda, K., Okamoto, T., Nakamae, E., Nishita, T., 1991. Photorealistic image synthesis for outdoor scenery. *The Visual Computer*. Vol. 7, pages 247-258.
- Maimone, M., Shafer, S., 1996. A taxonomy for stereo computer vision experiments. *ECCV workshop on performance characteristics of vision algorithms*. Pages 59-79.
- McCane, B., Novins, K., Crannitch, D., Galvin, B., 2001. On Benchmarking Optical Flow. *Computer Vision and Image Understanding*. Vol. 84, pages 126-143.
- Otte, M., Nagel, H., 1994. Optical Flow estimation: Advances and Comparisons. *Proc. 3<sup>rd</sup> European Conference on Computer Vision*. Pages 51-60.
- POV-Ray for windows, <http://www.povray.org>.
- Rokita, P., 1997. Simulating Poor Visibility Conditions Using Image Processing. *Real-Time Imaging*, 3, pages 275-281.
- Sim, R., Dudek, G., 1999. Learning and Evaluating Visual Features for Pose Estimation. *International Conference on Computer Vision*. Vol. 2.
- Stolkin, R., 2004. Combining observed and predicted data for robot vision in poor visibility. PhD thesis, Department of Mechanical Engineering, University College London.
- Stolkin, R., Hodgetts, M., Greig, A., 2000. An EM/E-MRF Strategy for Underwater Navigation. *Proc. 11<sup>th</sup> British Machine Vision Conference*.
- Tsai, R. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf tv camera and lenses. *IEEE Journal of Robotics and Automation*. Vol. 3(4), pages 323-344. 1987.
- Wunsch, P., Hirzinger, G., 1996. Registration of CAD-Models to Images by Iterative Inverse Perspective Matching. *Proceedings of the 13<sup>th</sup> International Conference on Pattern Recognition*. Pages 77-83.
- Zhang, Z., 1998. A Flexible New Technique for Camera Calibration. *Microsoft Research Technical Report, MSR-TR-98-71*.