

## Accurate and Simple Calibration of DLP Projector Systems

**Wilm, Jakob; Olesen, Oline Vinter; Larsen, Rasmus**

*Published in:*

Proceedings of SPIE, the International Society for Optical Engineering

*Link to article, DOI:*

[10.1117/12.2038687](https://doi.org/10.1117/12.2038687)

*Publication date:*

2014

[Link back to DTU Orbit](#)

*Citation (APA):*

Wilm, J., Olesen, O. V., & Larsen, R. (2014). Accurate and Simple Calibration of DLP Projector Systems. Proceedings of SPIE, the International Society for Optical Engineering, 8979, [897909]. DOI: 10.1117/12.2038687

## DTU Library

Technical Information Center of Denmark

---

### General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

# Accurate and Simple Calibration of DLP Projector Systems

Jakob Wilm<sup>1,2</sup> and Oline V. Olesen<sup>1,2</sup> and Rasmus Larsen<sup>1</sup>

<sup>1</sup>Department of Applied Mathematics and Computer Science,  
Technical University of Denmark,

Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby, Denmark;

<sup>2</sup>Department of Clinical Physiology, Nuclear Medicine & PET, Rigshospitalet,  
Copenhagen University Hospital, Denmark;

## ABSTRACT

Much work has been devoted to the calibration of optical cameras, and accurate and simple methods are now available which require only a small number of calibration targets. The problem of obtaining these parameters for light projectors has not been studied as extensively and most current methods require a camera and involve feature extraction from a known projected pattern. In this work we present a novel calibration technique for DLP Projector systems based on phase shifting profilometry projection onto a printed calibration target. In contrast to most current methods, the one presented here does not rely on an initial camera calibration, and so does not carry over the error into projector calibration. A radial interpolation scheme is used to convert features coordinates into projector space, thereby allowing for a very accurate procedure. This allows for highly accurate determination of parameters including lens distortion. Our implementation acquires printed planar calibration scenes in less than 1 s. This makes our method both fast and convenient. We evaluate our method in terms of reprojection errors and structured light image reconstruction quality.

**Keywords:** Projector calibration, radial basis functions, structured-light, phase shifting profilometry

## 1. INTRODUCTION

While camera calibration has been intensively studied, and state of the art methods have evolved, the calibration of projectors and camera projector systems is usually done using non-optimal methods, which sacrifice accuracy for speed or convenience. This is despite the fact that accurate determination of the optical parameters is of utmost importance in many applications within optical surface profilometry, stereolithography or visual display.

Current methods for projector calibration are usually relying on calibrated sensors to observe projection output and determine the appropriate parameters. This approach has the inherent disadvantage of propagating the sensor calibration error into the estimates of the projector parameters. A notable exception is the work of Moreno et al.,<sup>1</sup> which shares some properties with our proposed method.

While we do use a camera to determine world-projector relationships, we do not rely on the camera being calibrated. As such, no calibration error is propagated.

The pinhole-model is well established for digital cameras, and due to their optical similarities to most projectors, also applicable here. The model is usually further extended with distortion parameters to account for effects that don't follow projective geometry.

Our lens model contains ten parameters – horizontal and vertical focal lengths,  $f_x$ ,  $f_y$ , principal point coordinates  $c_x$ ,  $c_y$ , skew coefficient  $\alpha_c$  and lens distortion parameters  $k_1 \dots k_5$ . This is very similar to the widely accepted model of Heikkilä and Silvén,<sup>2</sup> and is the same as implemented in the popular camera calibration toolbox of Bouget<sup>3</sup> and in the OpenCV library.<sup>4</sup> Considering a world point in the camera's or projector's reference frame,  $\mathbf{Q} = [x_w, y_w, z_w]^T$ , the pin hole camera projection gives the normalized image coordinates as

$$\begin{bmatrix} u_n \\ v_n \end{bmatrix} = \begin{bmatrix} x_w/z_w \\ y_w/z_w \end{bmatrix}$$

---

Corresponding author: Jakob Wilm, E-mail jakw@dtu.dk, Telephone +45 45253716.

Spherical aberration gives rise to a radial distortion model according to

$$s(r) = 1 + k_1 r^2 + k_2 r^4 + k_5 r^6 \quad ,$$

with  $r^2 = x^2 + y^2$ , while the effects of imperfect lens alignment are termed tangential distortion, and modelled as

$$\mathbf{t}(x, y) = \begin{bmatrix} 2k_3 xy + k_4 (r^2 + 2x^2) \\ k_3 (r^2 + 2y^2) + 2k_4 xy \end{bmatrix} \quad .$$

This distortion effect is also called *decentering* or *thin prism* distortion, and was initially described by Brown.<sup>5</sup> The corrected normalised image coordinates of the point  $\begin{bmatrix} u_n \\ v_n \end{bmatrix}$  with radial and tangential distortion are given by

$$\begin{bmatrix} u_c \\ v_c \end{bmatrix} = s\left(\sqrt{u_n^2 + v_n^2}\right) \begin{bmatrix} u_n \\ v_n \end{bmatrix} + \mathbf{t}(u_n, v_n) \quad .$$

Finally, the pixel coordinates of said point are given by

$$\mathbf{q} = \begin{bmatrix} u_p \\ v_p \end{bmatrix} = \begin{bmatrix} f_x & \alpha_c f_x & c_x \\ 0 & f_y & c_y \end{bmatrix} \begin{bmatrix} u_c \\ v_c \\ 1 \end{bmatrix} = \mathbf{K} \begin{bmatrix} u_c \\ v_c \\ 1 \end{bmatrix}$$

Our method was developed for a new in-bore medical 3D scanner, which relies on accurate measurements, and requires a convenient calibration procedure that can be carried out by non-technical personal. To fulfil these goals, we have implemented a graphical user interface with visual feedback to ensure good coverage of the projector FOV and detection of feature point locations. This allows us to acquire a view of a calibration target in under one second, allowing for handheld operation of the projector or camera.

## 2. METHODS

Our method relies on fast acquisition of “projection sequences” with an industrial camera.

As mentioned earlier, our method does not rely on a calibrated camera. Instead we merely observe projection output in the presence of a planar checkerboard pattern. The projector encodes its pixels by means of *phase shifting profilometry* (PSP), and this allows us to establish world to projector plane correspondences.

We use a light grey checkerboard as the calibration target, which represents a trade off between robustness in corner detection and projector coordinate coding.

In recent iterations of their DLP technology, Texas Instruments has introduces DMDs with diagonal or *diamond* pixel arrays. The DLPC possesses a natural indexing of mirrors in this geometry, which has unique coordinates for every horizontal row and every other vertical column. Since we are interested in a linear mapping of pixel indices to position, we handle the diamond pixel array by assigning unique indices to every row and every column. With this definition, every other coordinate does indexes a “pseudopixel” – and in turn we encode only every other coordinate. See figure 2 for an illustration of the indexing convention for the DMD employed in our projector. It should be noted, that with this definition, calibration parameters and error estimates are expressed in units of our “pseudopixel”, which in physical dimensions corresponds to roughly half of a mirror diagonal.

After determining world coordinates in the projector coordinate system, we calibrate it like a camera using the method of Zhang.<sup>6</sup>

## 3. PHASE SHIFTING STRUCTURED LIGHT

Our encoding of projector coordinates into the projected output is realised with the gray level *phase shifting profilometry* (PSP) method pioneered by Hung.<sup>7</sup> While many methods exist for codification in structured light,<sup>8</sup> PSP allows for dense scene encoding with a low number of patterns.

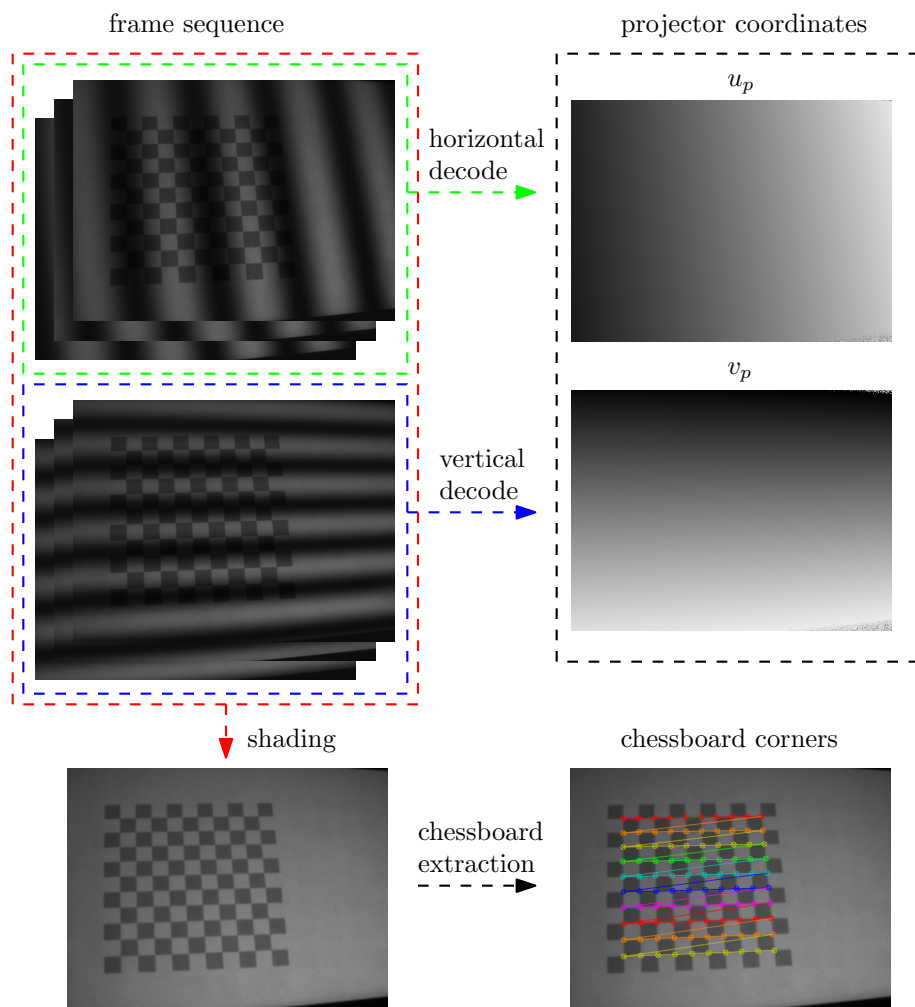


Figure 1. Flow diagram for the calibration procedure on one frame sequence. Horizontal and vertical encoding patterns are decoded to yield  $u_p$  and  $v_p$  maps. The patterns are also used to extract a general shading or intensity image, from which the chessboard corners are automatically extracted. These represent the checkerboard corners in the camera reference frame. By means of Gaussian radial basis functions, each set of corner coordinates is translated into projector coordinates.

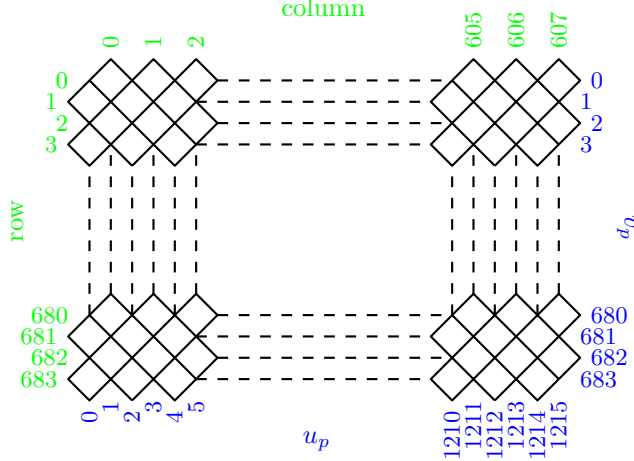


Figure 2. Diamond pixel DMD geometry for the DLP3000. The indexable screen coordinates are shown in green on top and left, while our pseudopixel coordinates are shown in blue on bottom and right. Note that in our pixel space, every other coordinate does not index a pixel, however, we do not need a dense pixel array for all subsequent steps.

Projector pixels are encoded first by their vertical ( $u_p$ ) and then by their horizontal ( $v_p$ ) coordinate. In the case of vertical encoding, the normalized pattern intensity may be described as

$$I_n^p(u_p, v_p) = \frac{1}{2} + \frac{1}{2} \cos\left(2\pi\left(\frac{n}{3} - v_p\right)\right),$$

where  $n$  indicates the pattern index,  $n \in 1 \dots 3$ . The intensity of the  $n$ 'th pattern as observed in the camera can be expressed as

$$I_n^c(u_c, v_c) = A^c + B^c \cos\left(\frac{2\pi n}{3} - \theta\right),$$

in which  $\theta$  denotes the phase of the sinusoidal sequence at that particular point.  $A^c$  is the intensity of the scene including ambient contributions, and  $B^c$  the intensity of reflected projector light. In terms of Fourier analysis,  $A^c$  can be considered the magnitude of the constant component, while  $B^c$  is the magnitude at the principle frequency, and  $\theta$  its phase. As such,  $\theta$  can easily be extracted by means of the fast fourier transform and scaled to the number of projector columns. The process for row codification is the same.

In order to reduce quantification error and noise, the number of encoded phases in PSP can be increased. The resulting map of  $\theta$  then needs to be unwrapped in order to remove the resulting phase ambiguity. A reliable way of unwrapping is using a single phase *phase cue* sequence. We employ this strategy of using six patterns for the horizontal and six patterns for the vertical encoding with 8 phases. Thus, the total number of projected patterns is 12 per calibration target pose. This gives us for every frame sequence the encoded projector row and column, and by calculating  $B^c$ , a general intensity or *shading* image, which we use for chessboard extraction (see Fig. 1).

#### 4. GAUSSIAN RADIAL BASIS FUNCTIONS

After horizontal and vertical frame sequence decoding and chessboard extraction, we seek to translate the chessboard corner coordinates from camera coordinates  $\mathbf{q}_c$  into projector coordinates  $\mathbf{q}_p$ . Collecting such coordinates from a number of chessboard corners from multiple views allows us to calibrate the projector by treating it as a camera and calibrate it such.

In our case, we wish to interpolate at the points determined from chessboard detection, e.g. at a sub pixel level. While the underlying function is expected to be close to linear, due to the effects of lens distortion, a more complex function underlies the observed data.

We treat the translation of every newly extracted chessboard corner as an interpolation problem. While the projector coordinates were estimated at regular sites, we do not employ gridded data interpolation. The reason

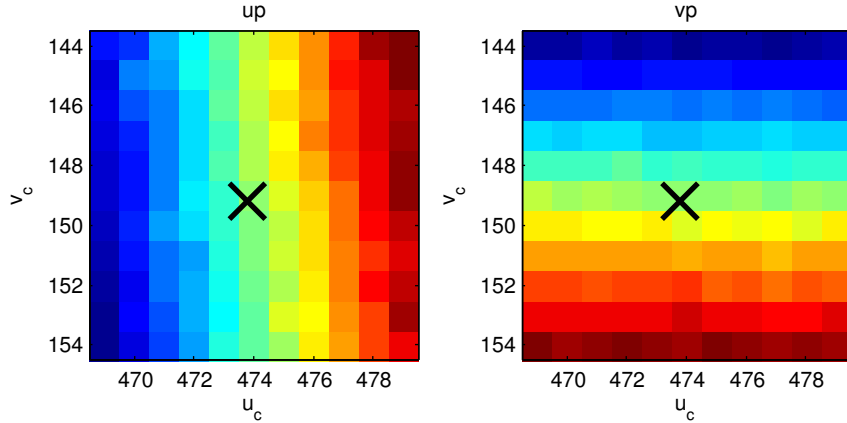


Figure 3. The interpolation problem exemplified on one chessboard corner location (black crosses). Left: the window of decoded projector columns. Right: same for projector rows.

is twofold; first we remove data by thresholding on the shading image, so we need to take care of missing data. Secondly, we wish to benefit from the advantages of using *radial basis function* (RBF) interpolators. Regular data interpolation is certainly the faster option, but in our application the complexity of RBFs does not become prohibitive.

In the RBF framework, a function is associated with each data point, and a regular set of equations solved to determine how each observation is explained as a linear combination of these functions. In order to ease our later analysis, we add a linear polynomial term  $P(\mathbf{x}_i)$ , such that the RBF will effectively describe the deviation from this global linear trend. To account for noisy data, we also add a regularization parameter  $k$  to our interpolator:

$$f_i = \sum_j \lambda_j \psi \|\mathbf{x}_i - \mathbf{x}_j\| + P(\mathbf{x}_i) + k\lambda_i \quad (1)$$

Common choices for  $\psi(r)$  include a Gaussian function,  $\psi(r) = \exp(-\alpha r^2)$  and the so-called Thin Plate Spline,  $\psi(r) = r^2 \log(r)$ . The choice of distance function influences the characteristics of the interpolator. Therefore we choose an RBF with a Gaussian distance function so as to give the highest weight to close neighbours while being robust against noise in the projector coordinate estimates. An influence region of around 100 neighbors seems reasonable, and correspondingly, as our interpolation sites are expressed in pixel coordinates, we have found good results with  $\alpha = 1/5$ .

All observations within a window around one detected chessboard corner are used to build the RBF interpolator. In practice this is done by forming the following equation from 1:

$$\begin{bmatrix} \Psi + kI & P^\top \\ P & \mathbf{0} \end{bmatrix} \begin{bmatrix} \lambda \\ c \end{bmatrix} = \begin{bmatrix} f \\ \mathbf{0} \end{bmatrix},$$

where  $\Psi$  contains the basis functions,  $P$  linear polynomial terms and  $k$  is the scalar parameter which controls regularization – i.e. the balance between the exact RBF interpolator and the simple linear polynomial, in this case a plane.

As mentioned, the underlying function is expected to have a linear trend, as without the effects of lens distortion, there is a direct perspective relationship between camera and projector coordinates.

Since we are operating with a regularized regression model, we can define a *hat* or *smoother* matrix  $\hat{H}$  according to

$$\hat{H} \begin{bmatrix} \Psi + kI & P^\top \\ P & \mathbf{0} \end{bmatrix} = [\Psi \quad P^\top],$$

Projector – TI LightCrafter				Camera – Point Grey FL3-FW-03S1M-C				
DMD Chip	Exposure	Bit Depth	Gamma	Image Sensor	Exposure	Bit Depth	Gamma	Gain
TI DLP3000	16.667 ms	8 bit	1.0	Sony ICX618 CCD	16.667 ms	8 bit	1.0	0 dB

Table 1. Data and default parameters for our DLP projector and camera during calibration.

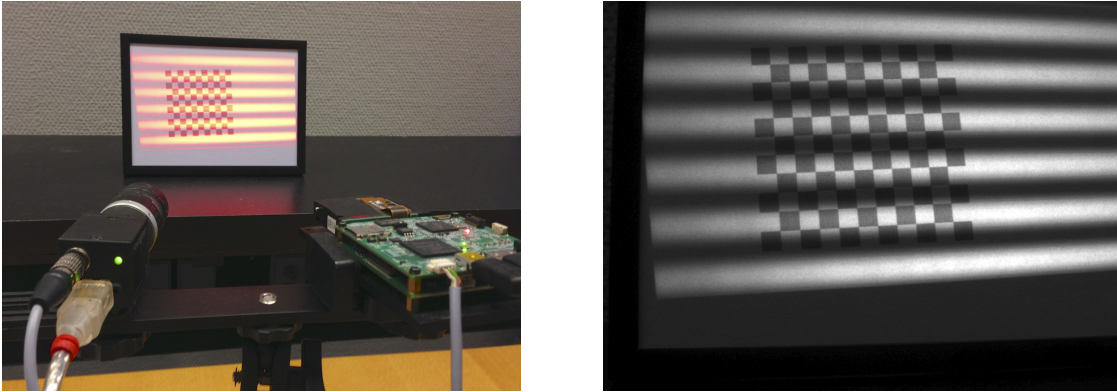


Figure 4. Left: Physical setup for our projector calibration method with camera (left) and projector (right). The calibration target is shown in the background. Right: corresponding frame captured by the camera.

and use the  $\text{Tr}(\hat{\mathbf{H}})$  as a measure of the *effective* degrees of freedom<sup>9</sup> of our model. We therefore choose  $k$  such that the effective number of degrees of freedom is 10, i.e. a little more than the 3 defining a plane, but much less than the 100 data points underlying each model.

## 5. IMPLEMENTATION

Our calibration setup is shown in figure 4. As seen, we have mounted the camera and projector in a fixed configuration, which allows us to also perform stereo calibration and triangulation world point coordinates using the structured light principle. Our calibration target is the checkerboard printed on a standard commercial laser printer using PostScript and spray glued onto the glass of a picture frame. This yields a highly accurate calibration target at a fraction of the cost of professionally manufactured offerings.

We use a LightCrafter DLP projector (Texas Instruments Inc.) with DLP3000 DMD (7.62 mm diagonal) and the FireWire industrial camera (Point Grey Research Inc.) FL3-FW-03S1M-C mono with Sony ICX618 CCD (4.00 mm diagonal). Table 1 shows default acquisition parameters. These values are chosen as a trade off between acquisition time and depth of field.

We have implemented our method as part of a complete structured light program in C++ utilizing the Qt, OpenCV and PCL (Point Cloud Library) libraries. See figure 5 for screenshots of the user interface. To enable fast pattern projection, we create a full screen OpenGL rendering context and upload the necessary patterns to the GPU as textures, which are rendered as screen filling quads. The camera interface uses libdc1394<sup>10</sup> to capture images from our FireWire industrial camera. In order to achieve a linear intensity response, we configure the projector to not use the typical spatiotemporal algorithms for visual image enhancement, and our camera to not perform gamma correction or signal amplification.

For each calibration pose, the corresponding series of captured frames is stored in the form of a frame sequence. Upon request, and when at least three frame sequences have been acquired, calibration is performed.

Checkerboard corners are automatically extracted using the appropriate functions in OpenCV’s *calib3d* package. This returns the desired image coordinates with subpixel accuracy. If checkerboard extraction is unsuccessful, the frame sequence is excluded from further processing. Otherwise, the frame sequence is decoded to yield projector coordinates and calibrate the projector like a camera using OpenCV’s built in functions.

Accurate capture of DLP projector output requires careful synchronisation of camera and projector exposure. While this is commonly realised using a trigger circuit, we opt for manual “software triggering”, i.e. image projection and image capture is initiated through software commands. To account for frame buffering in the GPU and the projector and latency in camera triggering, we impose a short pause in between frames.

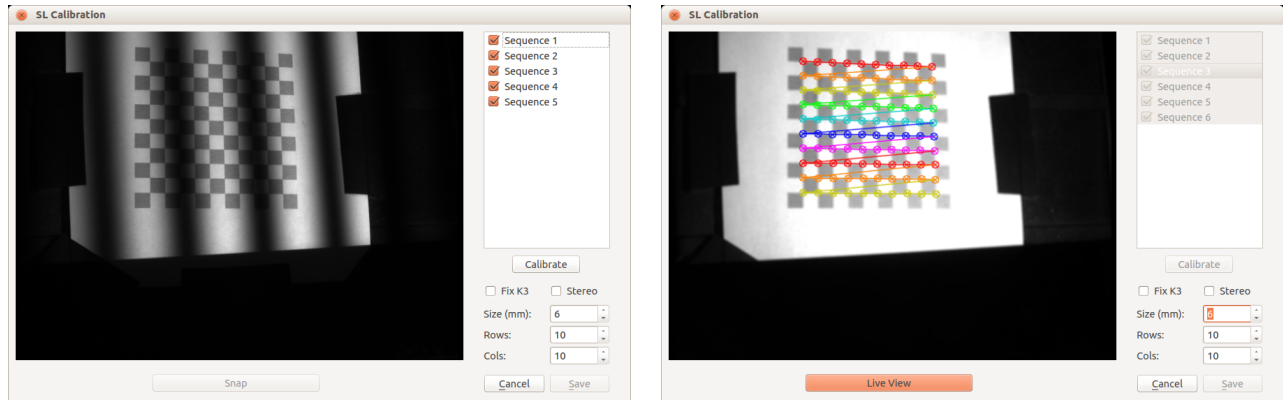


Figure 5. Video 1 Screenshots of our graphical user interface for calibration. Left: during calibration sequence acquisition. Right: during parameter estimation. <http://dx.doi.org/doi.number.goes.here>

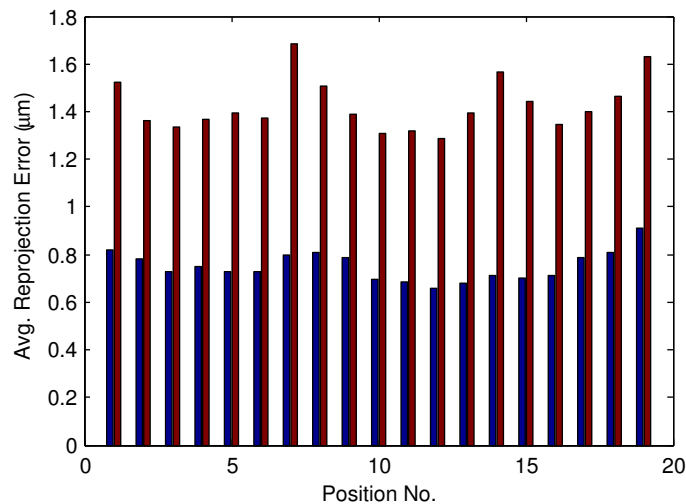


Figure 6. Average reprojection errors per pose for a calibration involving 18 calibration target poses for the camera (blue) and projector (red) respectively, measured in units of camera pixels and projector “pseudopixels” respectively.

## 6. RESULTS

Figure 6 shows reprojection error in a calibration involving 18 positions for the calibration target.

The individual reprojection errors are shown for the camera and for the projector on Fig. 7.

As the calibration target was moved through the intersecting FOVs of camera and projector three times, a positional dependence is apparent. This might indicate that the 5 parameter lens distortion model is not completely adequate. Low reprojection errors do not necessarily indicate a faithful calibration, as overfitting might explain low values. In this calibration case, a total of  $100 \times 18 \times 2 = 3600$  constraints are given for a rather low parametric model. No outliers are visible from the individual error plots in Fig. 7.

To measure the quality of 3D point reconstruction with our calibration method and correction for lens distortion, we have acquired structured light images of a spherical bowling ball that has been painted to a diffuse surface. The 2x3 PSP pattern strategy was used. We have then fitted a sphere to the resulting point cloud and assessed the deviations and the structure of the errors in a histogram, see Figure 8.

As can be seen, errors increase towards the edges, as the angle between camera axis and normal increases. The RMS of errors over the foreground is 0.57 mm. The error distribution is dominated mainly by triangulation error at the edges of the bowling ball.



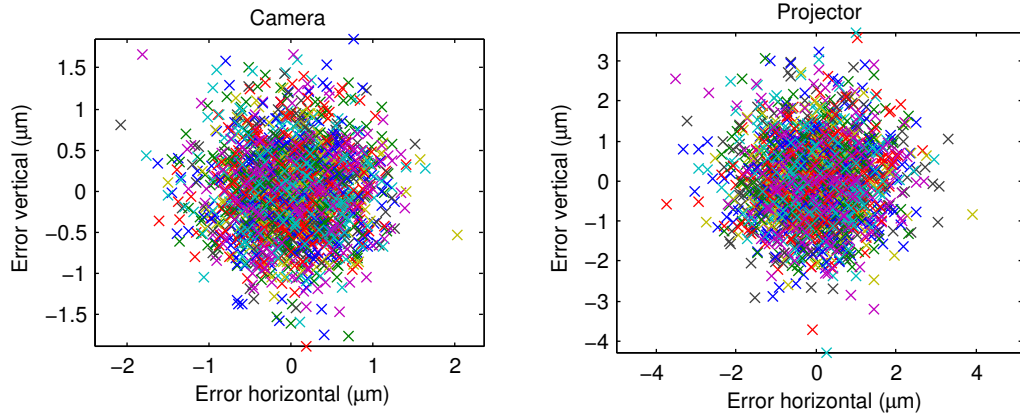


Figure 7. Individual reprojection errors with color coding corresponding to individual poses. Left: camera. Right: projector.

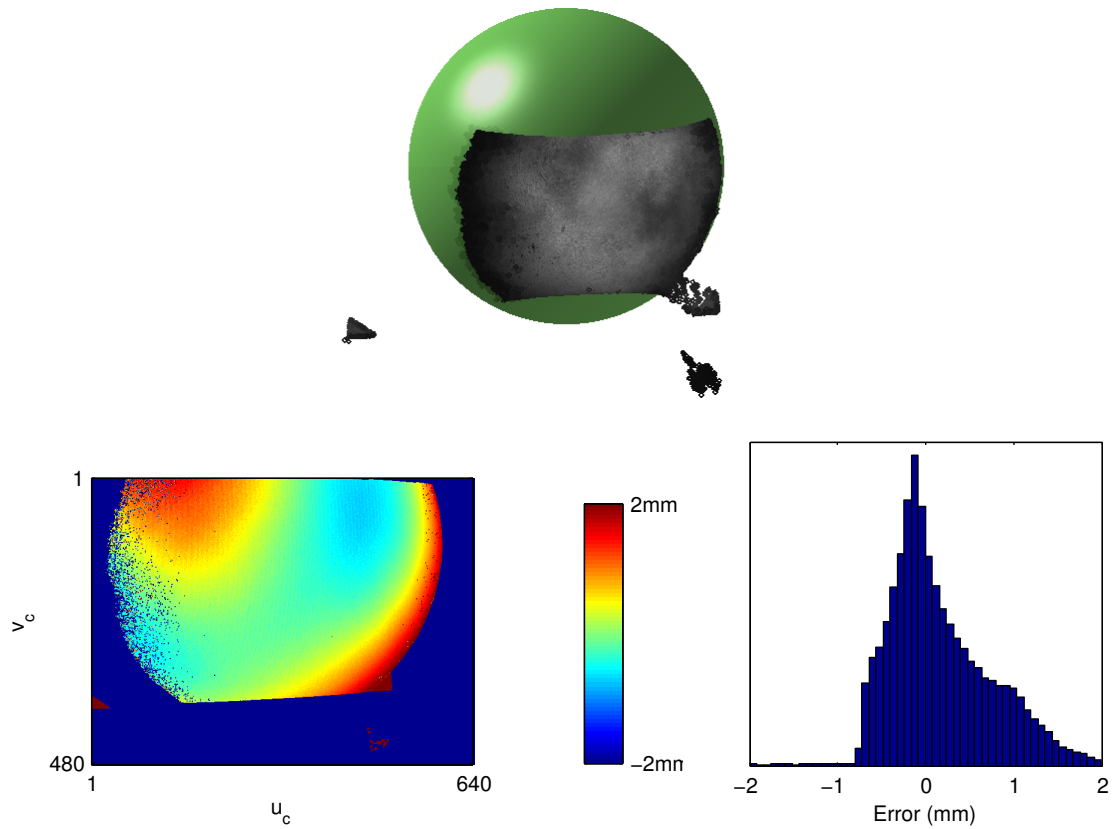


Figure 8. Structured light scan of a bowling ball. Top: sphere fitted to the triangulated data. Left: Plot of the error. Right: histogram of errors within the foreground. The RMS of errors is 0.57 mm.

## 7. CONCLUSIONS

We have shown a novel method and implementation of a convenient and accurate calibration method for light projectors. We utilize phase shifting profilometry for projector coordinate coding. Calibration points are translated into projector coordinates by means of Gaussian RBFs, which offers excellent trade-offs between accuracy and robustness. Visual feedback and direct camera/projector control enables operators to calibrate a light projector in a few minutes.

In addition, we determine projector lens distortion parameters with high accuracy and precision. Our method calibrates a camera at the same setting, and can perform a stereo calibration between projector and camera, which is particularly useful for single camera structured light applications.

Whilst most current methods require intricate setups or lengthy calibration procedures, we are able to calibrate very quick and accurately with only readily available equipment.

## REFERENCES

- [1] Moreno, D. and Taubin, G., “Simple, Accurate, and Robust Projector-Camera Calibration,” *2012 Second International Conference on 3D Imaging, Modeling, Processing, Visualization & Transmission*, 464–471 (October 2012).
- [2] Heikkila, J. and Silven, O., “Calibration procedure for short focal length off-the-shelf CCD cameras,” *Proceedings of 13th ICPR* **1** (1996).
- [3] Bouguet, J.-Y., “Camera Calibration Toolbox for Matlab.” [http://www.vision.caltech.edu/bouguetj/calib\\_doc/](http://www.vision.caltech.edu/bouguetj/calib_doc/)
- [4] Itseez, “OpenCV, Open Source Computer Vision.” <http://opencv.org/>
- [5] Brown, D. C., “Decentering Distortion of Lenses,” *Photometric Engineering* **32**, 444–462 (1966).
- [6] Zhang, Z., “Flexible camera calibration by viewing a plane from unknown orientations,” *The Proceedings of the ICCV 1999* (1999).
- [7] Hung, D. D., “3D scene modelling by sinusoid encoded illumination,” *Image and Vision Computing* **11**, 251–256 (June 1993).
- [8] Salvi, J., Pagès, J., and Batlle, J., “Pattern codification strategies in structured light systems,” *Pattern Recognition* **37**, 827–849 (April 2004).
- [9] Hastie, T., Tibshirani, R., and Friedman, J., “The elements of statistical learning data mining, inference, and prediction,” *Springer-Verlag* (2009).
- [10] Douchamps, D., “libdc1394: The API for IEEE1394.” <http://damien.douxchamps.net/ieee1394/libdc1394/>