Reflective Surface Reconstruction from Inverse Deflectometric Measurements

Dominik Penk[1], Roman Sturm[2], Lars Seifert[3], Marc Stamminger[1] and Günther Greiner[1]
[1]*Visual Computing Lab, Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Cauerstraße 11, Erlangen, Germany*
[2]*Rupp + Hubrach Optik GmbH, Von-Ketteler-Straße 1, Bamberg, Germany*
[3]*Fraunhofer IIS, Flugplatzstraße 75, Fürth, Germany*

Keywords: Reconstruction, Quality Control, Simulation, Numerical Optimization.

Abstract: Reconstructing reflective surfaces is a difficult task since most algorithms rely on photometric consistency between multiple views on the target object. However specular reflections are highly view dependent and thus violate this assumption. Previous work therefore often incorporates additional information, like polarization or the distortion of a known pattern, to perform specular surface reconstruction. We present a novel analysis by synthesis approach that defines an optimization problem using samples directly on the reconstructed surface. Based on this framework we describe two different setups for reconstruction, one using a line laser to create a reflection pattern and a second one, that uses point measurements to provide ray-measurement correspondences achieving improved accuracy.

1 INTRODUCTION

Above the age of 40 most people develop presbyopia, an eye disease caused by decreasing flexibility of the human eye lens. Focusing on close objects becomes increasingly more difficult. The most common tool to cope with this kind of defect is wearing glasses with varying refractive power. There are bifocals, trifocals and, nowadays most popular, so-called *progressive addition lenses* (PAL). They have a continuously varying refractive power, lower in the upper part of the lens (which is typically used to observe distant objects, e.g. when driving) and higher in the lower part of the lens (which is usually used for close objects, e.g. when reading).

The design of such lenses is highly sophisticated and the resulting geometry of the lens' surfaces is rather complex. The manufacturing is complicated and requires high precision CNC cutting and polishing machines. Quality control is very important, since small errors in the geometry may lead to high changes in refractive power. Currently used techniques to measure PALs have various problems: Either they are not sufficiently accurate, destructive, not fast enough or require expensive hardware. Therefore, they cannot be used in a production line to realize a complete quality control. Furthermore, they either only measure the curvature of one surface or infer a total lens power in transmission. For an as worn simulation however, a surface reconstruction of front and backside of the lens is necessary. We set out to develop an algorithm that directly computes the lens surface (either front or back face) and derive the optical properties from there.

In this paper we derive an analysis-by-synthesis approach to measure lenses based on recreating reflection patterns. We present two setups, one using a line laser. The laser sheet is reflected by the lens and generates a light curve on a screen that is recorded by a standard camera. Moving the lens through the laser generates a set of curves that are unique to each surface. We describe a novel optimization scheme, that from these curves directly reconstructs B-Spline coefficients defining the lens surface. Evaluation of this setup implies that reconstruction accuracy is primarily impeded by the continuous laser line prohibiting a clear matching of ray to line position. To overcome this hurdle, we propose the second setup in Sect. 4, which uses a dot line laser instead of a continuous line to produce single point measurements. These discrete points represent a sparser sampling of the reflection pattern, but also introduce a simple mapping from ray to curve point.

The main contribution of this paper is the in-

Penk, D., Sturm, R., Seifert, L., Stamminger, M. and Greiner, G.
Reflective Surface Reconstruction from Inverse Deflectometric Measurements.
DOI: 10.5220/0008968600760084
In *Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2020) - Volume 4: VISAPP*, pages 76-84
ISBN: 978-989-758-402-2; ISSN: 2184-4321
Copyright © 2022 by SCITEPRESS – Science and Technology Publications, Lda. All rights reserved

troduction of an intuitive and efficient optimization framework that

- directly determines surface parameters,
- has a simple analytic derivation supporting a fast optimization, and
- is easy to adapt to both setups.

Furthermore, we present results from real world measurements and simulated setups and discuss advantages and drawbacks for both presented setups.

2 RELATED WORK

While there are many well-established methods to measure/reconstruct surface geometry of diffuse objects, there are fewer techniques that are applicable to reflective or transparent objects. On the one hand there are contact based methods, so-called Coordinate Measuring Machines (Hocken and Pereira, 2011). The accuracy is satisfactory, but the time for precise measurements is high. Moreover, high precision CMMs are very costly. For measuring aspheric lenses/PALs optical based methods do a better job. A common approach is based on interferometry (Chamadoira et al.,). This method allows measuring the optical properties/effects of the lens (e.g. spherical curvature and astigmatism) at specified points. It is not possible to recover the actual surface geometry of the lens. Hence it does not allow variance analysis during production, i.e. comparing the fabricated lens with design data and checking compliance of fault tolerances. Another optical method for measuring is based on deflectometry (Knauer et al., ; Kaminski, 2008; Knauer, 2006; Werling et al., 2007). One or two cameras capture the mirror image of a regular pattern (typically a sinusoidal one). Then the methods infer a normal field that leads to the distortion of the regular pattern. This method is very precise and can recover the geometry of the reflective surface by integration within reasonable time. When it is applied to PALs, the back surface must be blackened during the measurement. Clearly, this prohibits the use of the lens after the measurement, i.e. this method is destructive and cannot be used in a production line.

The method proposed by (Wedowski et al., 2012) shares a similar setup to the one we propose, however an initial reconstruction is required. In a second step they estimate finer details on the object by estimating a normal map from the reflection pattern.

To specify the geometry of PALs different mathematical models are used in the industry. Most popular are Zernicke polynomials (*ZP*) and tensor product B-Splines. We use the latter description. First of all because the samples we measured with our approach where designed accordingly (Loos et al., 1998). Moreover, to represent lenses with highly varying refractive power, a *ZP* of high degree (up to 20) is required, which causes stability issues for numeric surface evaluation. Using tensor product B-splines, the number of control points can be increased to gain sufficient variability in the geometry, the polynomial degree remains low and surface evaluation is numerically reliable.

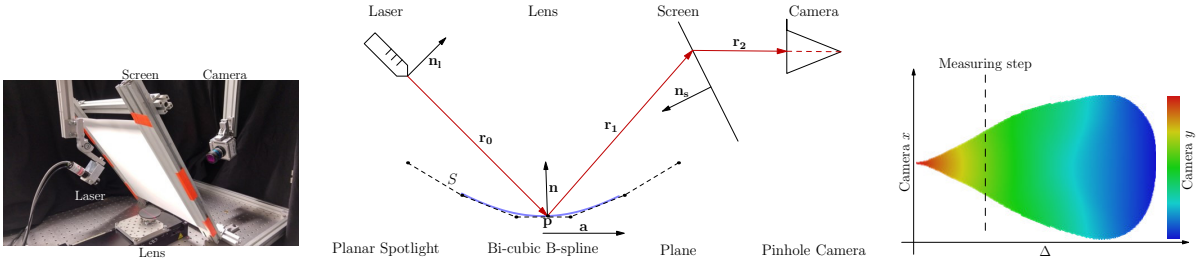
3 LINE MEASUREMENT

3.1 Setup

Our measuring arrangement is depicted in Fig. 1a and consists of a line laser, a screen and a standard camera. During a scan, we move the measured object with constant velocity and capture the resulting reflection pattern on the screen. For each frame in the acquired video stream, we detect the reflection curve. To ease the following computations, we assume that the curve has an explicit representation and find the y-coordinate of the reflection curve for each camera column. We store these values as columns of an image, as depicted in Fig. 1c. The resulting image is parameterized by the frame index Δ and the camera column; pixel values denote the y-coordinate of the reflection curve in frame Δ. We tested our setup with a variety of different lenses and were always able to find an arrangement that produced such explicit reflection curves. We call this new measuring method *Inverse Deflectometry* referring to the related classic *Deflectometry* (Knauer et al., ; Kaminski, 2008; Knauer, 2006; Werling et al., 2007). In contrast to this method we do not project a dense pattern onto the measuring object and directly observe the distortion on the surface. We instead capture the reflection pattern on a screen and infer the shape from this indirect measurement. This enables us to clearly separate the reflections from different sides of the lens enabling us to conduct a nondestructive measurement. For the setups presented in this paper we filtered the camera images to remove backface reflections prior to line extraction. Another important difference is that our method generates an actual surface reconstruction whereas the direct deflectometirc method produces a normal field.

3.2 Scan Simulation

Our reconstruction method is an analysis-by-synthesis approach. We use a parametrized geometric



(a) Measurement setup (b) Measurement setup sketch (c) Sample measurement result

Figure 1: Subfigure (a) depicts the proposed setup. (b) is a sketch of the same setup with annotated parts. In the top row the real-world objects are listed, on the bottom we show the corresponding models used for simulation. (c) shows an exemplary result of a measuring drive. The individual measurement steps represent a single column in the image with the y-coordinate of the reflection lines represented by colors.

surface model to simulate the acquisition process and optimize the surface parameters to reproduce the presented measurement. We model the screen as a plane and use an extended pinhole model to simulate the camera and account for lens distortion during acquisition by undistorting the camera images before extracting measurement data.

The surface description is a vital part of the reconstruction and should be chosen carefully to ensure a high-quality reconstruction while simultaneously retaining decent runtimes. In particular, the model should require as few codependent parameters as possible. Besides this, our use case requires that the mean curvature is well defined on the entire surface.

Based on these preconditions we choose an explicit uniform bi-cubic B-spline S (Farin, 2002) as the geometric surface model. S is defined by height values h_{ij} arranged on a regular $m \times n$ grid $\mathbf{h}$ and two associated knot-vectors. To form an explicit representation of the surface, the knots are chosen such that the xy-coordinates are always equivalent to uv-coordinates on the B-spline domain. The z-coordinate of a surface point is given by a convex combination of the height values:

$$S(\mathbf{uv},\mathbf{h}) = \sum_{i,j} w_{ij}(\mathbf{uv}) h_{ij} = \mathbf{w}(\mathbf{uv})^T \mathbf{h} \qquad (1)$$

$w_{ij}(u,v)$ is the product of the bi-cubic B-spline basis functions described in (Farin, 2002). We drop the arguments for brevity if they are unambiguous. Note that the local support of the B-spline basis functions implies that only 16 coefficients are nonzero for any point on the surface. Fig. 1b shows a sketch of the proposed setup including real world hardware and associated virtual models used during simulation.

3.3 Optimization

Let R be a subset of all rays emitted by the laser. By tracing one of these rays $r \in R$ through the measuring setup, using the surface S, parameterized by the control points $\mathbf{h}$, and a lens offset Δ, we find the corresponding simulated measurement $\mathbf{q}(r,\mathbf{h})$ on the camera chip. Using this simulation, we can formulate an intuitive error term:

$$E_{\mathrm{R}}(\mathbf{h},M) = \frac{1}{2} \sum_{r \in R} |M(\mathbf{q}(r,\mathbf{h})_x, \Delta) - \mathbf{q}(r,\mathbf{h})_y|_\gamma \qquad (2)$$

Here $M(row, column)$ is an image access and $|\cdot|_\gamma$ is a stable norm-like function (e.g. Huber-norm).

There are however some major drawbacks to this formulation. During optimization each ray has to be intersected with the surface. For a general bi-cubic B-spline this is most efficiently done with an iterative process, which is still relatively slow, and an analytical derivation of this process is nontrivial. Finally, many of the rays may not even hit the surface and therefore do not add any information to the optimization.

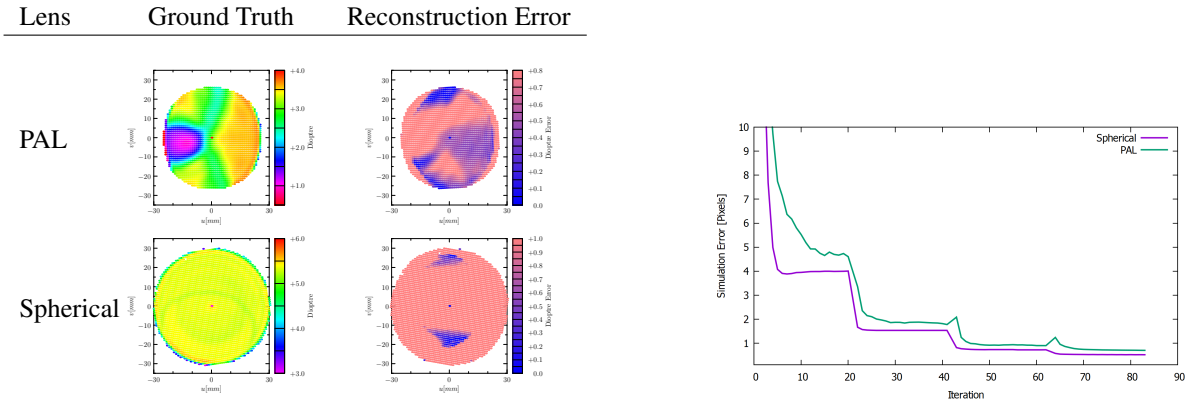
We thus propose a novel optimization scheme, that measures the error by sampling the surface S directly. Since this is the domain of the optimization target we will only generate meaningful residuals and do not waste computation time for ray surface intersection. The line laser emits a planar bundle of rays and we can easily compute the advance Δ and ray r, required for simulating the measurement, given a sample $\mathbf{uv}$ on S:

$$\Delta(\mathbf{uv}) = -\frac{\mathbf{n}_l^T S(\mathbf{uv}) + d_l}{\mathbf{n}_l^T \mathbf{a}} \qquad (3)$$

$$r(\mathbf{uv}) = (\mathbf{o}_l, S(\mathbf{uv}) - \mathbf{o}_l) \qquad (4)$$

Where $\mathbf{o}_l, \mathbf{n}_l$ and d_l define the laser and $\mathbf{a}$ is the displacement vector of the lens between two consecutive frames. The error term using surface samples is very similar to Eq. 2:

$$E_{\mathrm{S}}(\mathbf{h},M) = \frac{1}{2} \sum_{\mathbf{uv}} |M(\mathbf{q}(r,\mathbf{h})_x, \Delta - \mathbf{q}(r,\mathbf{h})_y|_\gamma \qquad (5)$$



(a) Ground truth error.

(b) Simulation error.

Figure 2: (a) displays the divergence of reconstructed and ground truth diopter over the surface. (b) depicts the value of the objective function without the smoothing term.

Since lenses are smooth, we encourage smoothness by adding the thin plate energy term to the objective function:

$$\Phi_{\mathrm{TP}}(\mathbf{h}) = \int_{\Omega} \left(S_{uu}^2(\mathbf{uv}) + 2S_{uv}^2(\mathbf{uv}) + S_{vv}(\mathbf{uv})^2\right) d\mathbf{uv} \tag{6}$$

Here S_{uu}, S_{uv} and S_{vv} are the second derivatives of the bi-cubic B-spline S. This energy function is often used when working with splines, e.g. for mesh fairing (Greiner, 1994) or data approximation (Bookstein, 1989). Combining this regularization with the original simulation error yields the final objective function:

$$\underset{\mathbf{h}}{\operatorname{argmin}}\ E_{\mathrm{S}} + \lambda_{\mathrm{TP}}\Phi_{\mathrm{TP}} \tag{7}$$

We can solve this optimization problem using standard nonlinear solvers (e.g. LM (More, 1977)) which require partial derivatives for the objective function. Eq. 3 is based on ray tracing which is a recursive method. For each ray intersection we only require knowledge of the incoming ray and some parameters specific to the current intersection (e.g. surface normal for reflection or implicit plane equation for ray-plane intersection). Essentially, the new ray is given by $r_{i+1} = f(r_i, \mathbf{h})$ and we can apply the chain rule to obtain the derivative:

$$\frac{\partial r_{i+1}}{\partial \mathbf{h}} = \frac{\partial f}{\partial r_i}\frac{\partial r_i}{\partial \mathbf{h}} \tag{8}$$

To provide all data required for optimization, we extend the classic ray definition by the partial derivatives of origin and direction. Eq. 3 implies that the lens offset Δ depends on $\mathbf{h}$. We therefore add it to the ray data leading to a sextuple describing a ray r:

$$r = \left(\mathbf{o}, \mathbf{d}, \Delta, \frac{\partial \mathbf{o}}{\partial \mathbf{h}}, \frac{\partial \mathbf{d}}{\partial \mathbf{h}}, \frac{\partial \Delta}{\partial \mathbf{h}}\right) \tag{9}$$

Directly sampling the surface implies that the derivative of the initial ray $\frac{\partial r_0}{\partial \mathbf{h}}$ solely depends on the B-spline weights $\mathbf{w}$:

$$\frac{\partial r_0}{\partial \mathbf{h}}(\mathbf{uv}) = (\mathbf{0}, \mathbf{w}) \tag{10}$$

The derivative is constant throughout optimization and we therefore can precompute the structure of the sparse Jacobian matrix. This is another advantage of the proposed sampling strategy over using a fixed set of rays. If the optimization is set up according to Eq. 2 each residual will produce a different surface sample in each iteration potentially changing the influenced control points and therefore the structure of the Jacobian.

3.4 Results

To demonstrate our method we show the results of two measured lenses: a simple spherical lens and a custom designed PAL. Fig. 2a shows the resulting deviations from the ground truth captured via the phase measuring deflectometric method (PMD) (Knauer, 2006). We blacked and roughed the front faces of the glasses to capture the ground truth. For the PAL we started optimization from a sphere with a radius corresponding to the far sight region of the lens. For the spherical lens we started from a plane.

The presented results show that the optimization converges to a low simulation error (see Fig. 2b), but the recovered surface is not sufficiently accurate. In the following we analyze the reasons for this insufficient accuracy.

A major hurdle for most reconstruction methods based on reflection patterns is the height ambiguity, as described in (Knauer, 2006). The problem is visualized in Fig. 3: Given a single ray r and its mea-

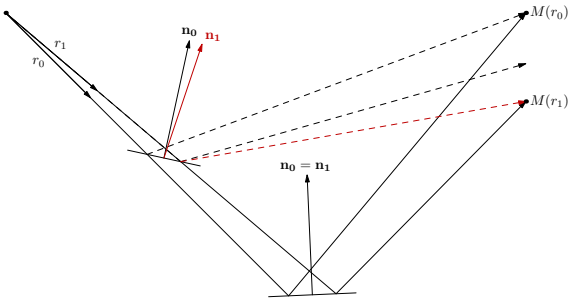


Figure 3: Depiction of the height problem and how it is resolved with a parameterized surface. Along a single ray (e.g. r_0) we can always derive a normal to reflect it towards the measurement. For two rays however, there is only one feasible plane.

surement $M(r)$ we can choose an arbitrary point on r and derive a normal that reflects the ray towards its assigned target. This ambiguity is a difficult challenge for many deflectometric methods since the number of solutions is infinite and the uncoupled ray measurements have to be combined without physical reasoning.

Our method, in contrast, by construction introduces plausible dependencies between rays by intersecting them with a common surface. Fig. 3 shows that this implicitly solves the height ambiguity for the simple case of a planar surface. We need at least two rays to construct a well determined system since each ray creates two residuals in Eq. 5 and a plane is defined by four parameters. This reasoning transfers to bi-cubic B-spline surfaces used in our method, assuming that the B-spline is sufficiently variable to describe the actual lens. Since the B-spline representation has a local support it is important to sample the entire domain to create a well-defined optimization problem. We directly sample the surface to generate residuals for the optimization and therefore can ensure this easily.

The second source of ambivalence in our setup is the matching of rays to point along the measured curves. Inspecting the simulated reflection lines using the final optimization result implies that this is indeed a major problem. The optimization tends to generate reflection curves that are shorter than the measured lines and only cover a subsegment. As seen in Fig. 2b the simulation error still diminishes since Eq. 5 enforces that the simulated line is on the measured line but does not require complete coverage. Experiments to add an error term to enforce total line coverage were not successful. We therefore propose to introduce a mapping from rays, produced by the laser, to a position on the reflected line.

(Wedowski et al., 2012) follow a similar idea where they directly map the simulated to measured reflection. Since they start their reconstruction from an initial photometric scan, they assume that the simulation is already very similar to the measurement and linearly map line segments between distinctive points. We strife to provide a reconstruction method without any specific initial guess. In the following we therefore change our first setup slightly to provide the ray-measurement matching directly.

4 DOT LINE MEASUREMENT

4.1 Setup Modification

There are many possibilities to introduce ray-measurement correspondence to the measurement. We propose to replace the continuous line laser with a laser producing a dotted line. Many of these lasers provide a recognizable center ray (e.g. using a bigger radius) and we can from there infer the remaining points by simply counting the distance to the center ray. The measurement output changes slightly: Instead of a height field of stacked lines we use an image of size $|dots| \times |images|$ where each entry stores the pixel position where the corresponding dot is detected in by camera. See Fig. 4 for an exemplary measurement. We update the simulation error to accommodate for the new data:

$$\hat{E}_S(\mathbf{h}, M) = \frac{1}{2} \sum_{\mathbf{uv}} d\left(M(r, \Delta) - \mathbf{q}(r, \mathbf{h})\right) \qquad (11)$$

Where $M(row, column)$ is once again an image access and $d(\cdot)$ is an appropriate distance function (e.g. squared L_2 norm). Given this new measurement data we can now also work with loops, assuming the measurement can handle them, since the pixel coordinates are now completely separate from the measurement domain.

The modified setup replaces continuous line samples with a discrete sampling of the reflection pattern. Therefore, the laser rays, present in the real-world measurement, also samples the lens surface at

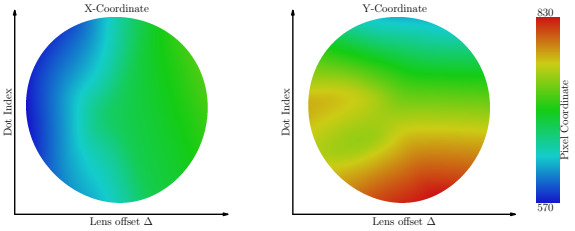
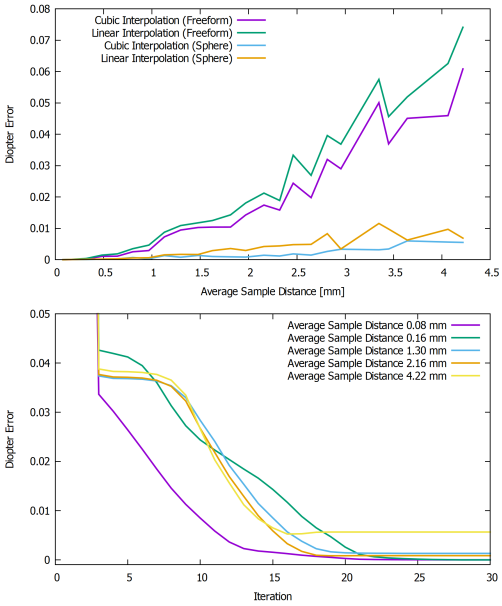
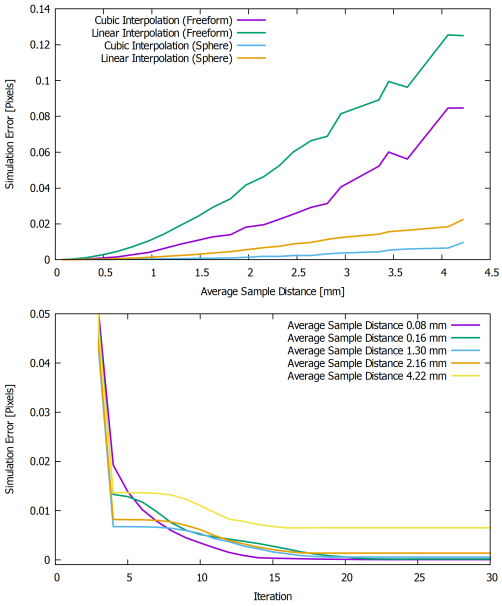


Figure 4: An exemplary new measurement with pixel coordinates displayed separately and color-coded.



(a) Mean absolute diopter difference to ground truth.



(b) Simulation error E_S as defined in Eq. 11.

Figure 5: Ground truth and simulation error for a variety of lenses and sample rates.

discrete positions. Our optimization chooses random points on the spline domain and therefore will frequently trace rays not present in the measurement. To generate an accurate reconstruction, it is paramount to find a good estimate of the reflection pattern for these unseen rays. Depending on the distance between consecutive measurement points commonly used linear interpolation might not be sufficient. We instead opted for a bicubic interpolation of the measurement since the reconstructed lenses are very smooth and therefore also create a smooth reflection pattern. As we will show in the following chapter this interpolation method produces better results, especially for coarser sample patterns compared to linear interpolation.

4.2 Results

To thoroughly evaluate the behavior of our method we used a simulated environment where we can easily test many different parameters of the setup. To evaluate the behavior of the reconstruction with different sampling rates we simulated several PAL designs (denoted as freeform) and spherical lenses ranging from negative to highly positive diopter values. The reconstruction progress always started from a plane. Results are summarized in the plots of Fig. 5.

We successively increased the angle between rays produced by the laser which leads to increasing spread of scanned points on the lens. The sample distance ranges from an average of 0.08 mm to 4.22 mm. As expected, we can use a much wider spread of laser rays for spherical lenses compared to PALs. For the latter surfaces we could potentially loose details that are small enough to fit in the gap between the discrete data points. Note that the simulation error for PAL surfaces is generally higher. The reason is twofold, first the smoothing term Eq. 6 prevents overfitting to the measurement. On the other hand, the reflection curve produced by a PAL is less smooth in comparison to a spherical lens resulting in a larger interpolation error.

Fig. 6 shows some exemplary reconstructions starting from a plane. The sample rate was chosen to generate an average dot sample distance of 2mm. Larger deviations from the ground truth are primarily located at fringes of the lens. To represent the entire lens surface with the B-spline, we chose a domain larger than the actual lens and cut out the spher-

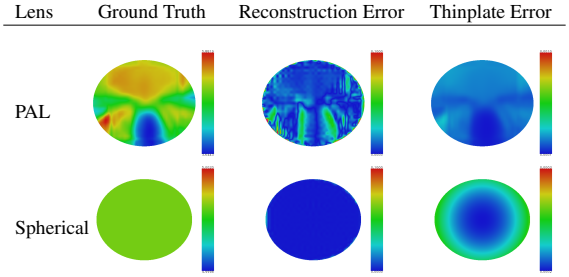


Figure 6: Reconstruction results for exemplary lenses. The reconstruction error ist the absolute difference to the ground truth and clamped to 0.1 dioptre.

ical shape. This implies that fewer residuals are created for control points near the border of the lens and the smoothing term has a larger influence leading to worse reconstruction. However real measurements tend to create very noisy and inaccurate measurements near the edge of the lens. These error result from inter reflections or rough surface on the edges, such that we decided not to reduce the influence of the smoothing term towards the fringes.

As mentioned in Sect. 4.1 we assume that the interpolation of the sparse measurements has a profound impact on the final reconstruction. To verify this, we compared the bicubic and bilinear interpolation of M for different sample rates. The first method results in a smooth function even at pixel borders, whereas the linear interpolation can cause G1-discontinuities. The top plot in Fig. 5a shows that the bicubic interpolation is generally the superior interpolation method especially for freeform surfaces with a lower sample rate. For higher sample rates however, the linear interpolation is also a good approximation.

Finally, we study the convergence rate of the optimization procedure in the second row of Fig. 5. The simulation and ground truth error behave similar and only change insignificantly for the first fifteen to twenty iterations for the simulated objects.

5 CONCLUSION AND FUTURE WORK

We presented two setups to reconstruct a lens surface from its reflection pattern. We have shown that a continuous line, used in the first setup, produces reconstructions of inadequate accuracy for quality control. However, the proposed optimization framework and setup was adapted to explicitly provide ray-measurement correspondences. The optimization framework is agnostic to the actual measurement technique, if a mapping from the surface point to the space of the light source is available. In future work we will try different types of lasers that directly encode the angle along the ray (e.g. using phase encoding) and evaluate if the denser measurements significantly impact reconstruction quality.

In this paper we explicitly filtered secondary reflections to stick to a simple light path with a single reflection at the initial surface. We therefore require two scans to fully capture a single lens. The presented framework can be extended easily to more complex paths with multiple surface interactions. We will incorporate this into our method to simultaneously capture both sides of a PAL in a single measurement.

Although the reconstruction time of around 2 minutes is short enough to include the method as a quality assurance tool into an existing production line, we would like to speed up the process. Since the simulations of the individual laser rays, done during residual computation, are independent, we anticipate a meaningful speedup by implementing a GPU based version of our algorithm.

ACKNOWLEDGEMENTS

This research is funded by Bayerische Forschungsstiftung ”Schritthaltende 3D-Rekonstruction und -Analys (AZ-1184-15)” (For3D). The lenses and PALs are provided by our project partners Rupp+Hubrach Brillenglas.

REFERENCES

Agarwal, S., Mierle, K., and Others. Ceres solver. http://ceres-solver.org.

Böhm, W. (1977). Cubic b-spline curves and surfaces in computer aided geometric design. *Computing*, 19(1):29–34.

Bookstein, F. L. (1989). Principal warps: thin-plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6):567–585.

Chamadoira, S., Blendowske, R., and Acosta, E. Progressive addition lens measurement by point diffraction interferometry. *OPTOMETRY AND VISION SCIENCE*, 89(10):15321542.

Farin, G. (2002). *Curves and Surfaces for CAGD: A Practical Guide*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 5th edition.

Greiner, G. (1994). Variational Design and Fairing of Spline Surfaces. *Computer Graphics Forum*, 13(3):143–154.

Halstead, M. A., Barsky, B. A., Klein, S. A., and Mandell, R. B. (1996). Reconstructing curved surfaces from specular reflection patterns using spline surface fitting of normals. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH ’96, pages 335–342, New York, NY, USA. ACM.

Hocken, R. and Pereira, P. (2011). *Coordinate Measuring Machines and Systems, Second Edition*. CRC Press.

Kaminski, J. (2008). *Geometrische Rekonstruktion spiegelnder Oberflächen aus deflektometrischen Messdaten*. PhD thesis, FAU Erlangen Nürnberg.

Kaminski, J., Lowitzsch, S., Knauer, M. C., and Häusler, G. (2006). Full-field shape measurement of specular surfaces. In Osten, W., editor, *Fringe 2005*, pages 372–379, Berlin, Heidelberg. Springer Berlin Heidelberg.

Knauer, M. (2006). *Absolute Phasenmessende Deflektometrie*. PhD thesis, FAU Erlangen Nürnberg.

Knauer, M., Kaminski, J., and Häusler, G. Phase measuring deflectometry: a new approach to measure specular free-form surfaces. In editor, editor, *Proceedings of SPIE - The International Society for Optical Engineering*, volume 5457.

Loos, J., Greiner, G., and Seidel, H.-P. (1998). A variational approach to progressive lens design. *Computer-Aided Design*, 30(8):595 – 602.

More, J. (1977). Levenberg–marquardt algorithm: implementation and theory. *Numerical analysis*, 630.

Moreno, D. and Taubin, G. (2012). Simple, accurate, and robust projector-camera calibration. In *2012 Second International Conference on 3D Imaging, Modeling, Processing, Visualization Transmission*, pages 464–471.

Rahmann, S. and Canterakis, N. (2001). Reconstruction of specular surfaces using polarization imaging. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–149–I–155 vol.1.

Richter, C., Kurz, M., Knauer, M., and Faber, C. (2009). Machine-integrated Measurement of Specular Free-formed Surfaces Using Phase-measuring Deflectometry, vol. 6. *EUSPEN 2009*.

Sorkine, O. (2005). Laplacian Mesh Processing. In Chrysanthou, Y. and Magnor, M., editors, *Eurographics 2005 - State of the Art Reports*. The Eurographics Association.

Tan, C.-T., Chan, Y.-S., Lin, Z.-C., and Chiu, M.-H. (2011). Angle-deviation optical profilometer. *Chin. Opt. Lett.*, 9(1):011201.

Wang, S.-W., Shih, Z.-C., and Chang, R.-C. (2002). An efficient and stable ray tracing algorithm for parametric surfaces. *J. Inf. Sci. Eng.*, pages 541–561.

Wedowski, R., Atkinson, G., Smith, M., and Smith, L. (2012). Dynamic deflectometry: A novel approach for the on-line reconstruction of specular freeform surfaces. *Optics and Lasers in Engineering*, 50:1765–1778.

Werling, S., Balzer, J., and Beyerer, J. (2007). A new approach for specular surface reconstruction using deflectometric methods. In *Informatik 2007 - Informatik trifft Logistik. Bd.1. Hrsg.: R. Koschke*, volume 109 of *GI-Edition / Proceedings*, pages 44–48. Gesellschaft für Informatik, Bonn.

APPENDIX

The integral part of our reconstruction method is tracing a single ray through the acquisition geometry and the derivative of this path according to the B-spline height values **h**. In this appendix we present the formulas and derivatives needed to construct most paths present in a real-world setup. We will use the shorthand f_x for $\frac{\partial f}{\partial x}$ for brevity in this appendix.

Both reflection and refraction require the normalised normal at a point (u,v) on the B-spline domain. The normal is easily derived since we chose an explicit version of the B-spline:

$$\mathbf{n} = \frac{\begin{pmatrix}-S_u & -S_v & 1\end{pmatrix}^T}{\left|\left|\begin{pmatrix}-S_u & -S_v & 1\end{pmatrix}\right|\right|} \tag{12}$$

We define $l = \left|\left|\begin{pmatrix}-S_u & -S_v & 1\end{pmatrix}\right|\right|$, $\mathbf{w_u}$ and $\mathbf{w_v}$ are the derivatives of the B-spline basis functions with respect to the indices.

$$\mathbf{n_h} = \frac{\begin{pmatrix}\mathbf{w}_u^T \\ \mathbf{w}_v^T \\ 0\end{pmatrix} l + \begin{pmatrix}S_u^T \\ S_v^T \\ 1\end{pmatrix} (S_u\mathbf{w}_u + S_v\mathbf{w}_v)}{l^2} \tag{13}$$

After a surface intersection the incoming ray **i** either refracts into the surface (**t**) or gets reflected (**r**) as shown in Fig. 7. In reality both cases will occur but since only the shape of the resulting pattern is of interest, we define a fix interaction type during optimization. The cosine of the incident angle and its' derivation is given by

$$\begin{aligned}\cos(\theta_i) &= -\mathbf{i}^T\mathbf{n} \\ \cos(\theta_i)_\mathbf{h} &= -\mathbf{i}^T\mathbf{n_h} - \mathbf{n}^T\mathbf{i_h}\end{aligned} \tag{14}$$

From this the reflection direction is easily computed:

$$\begin{aligned}\mathbf{r} &= \mathbf{i} + 2\cos(\theta_i)\mathbf{n} \\ \mathbf{r_h} &= \mathbf{i_h} + 2\left(\cos(\theta_i)_\mathbf{h}\mathbf{n} + \cos(\theta_i)\mathbf{n_h}\right)\end{aligned} \tag{15}$$

The refractive indices of the two neighboring media at the surface are needed to compute the refraction ray,

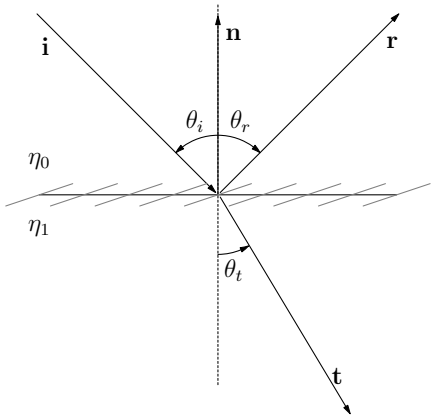


Figure 7: Refraction and Reflection.

which has the following form:

$$\begin{aligned}
\mathbf{t} &= \frac{\eta_1}{\eta_2}\mathbf{i} + \left(\frac{\eta_1}{\eta_2}\cos(\theta_i) - \sqrt{1-\sin^2(\theta_t)}\right)\mathbf{n} \\
\mathbf{t_h} &= \frac{\eta_1}{\eta_2}\mathbf{i_h} \\
&+ \left(\frac{\eta_1}{\eta_2}\cos(\theta_i)_\mathbf{h} + \frac{\sin^2(\theta_t)}{\sqrt{1-\sin^2(\theta_t)}}\right)\mathbf{n} \\
&+ \left(\frac{\eta_1}{\eta_2}\cos(\theta_i) - \sqrt{1-\sin^2(\theta_t)}\right)\mathbf{n_h}
\end{aligned} \tag{16}$$

with

$$\begin{aligned}
\sin^2(\theta_t) &= \left(\frac{\eta_1}{\eta_2}\right)^2 (1-\cos^2(\theta_i)) \\
\sin^2(\theta_t)_\mathbf{h} &= -2\left(\frac{\eta_1}{\eta_2}\right)^2 \cos\theta_i \cos(\theta_i)_\mathbf{h}
\end{aligned} \tag{17}$$

After the ray interacted with the lens, we have to model the final two steps along the light path. The first one is the point on the screen defined by the plane

$$\mathbf{n}_S^T\mathbf{p} + d_S = 0 \tag{18}$$

where $\mathbf{n}_S$ and $\mathbf{p}$ a point on the plane. If we insert a point on the ray r into this equation we can compute the intersection point $\mathbf{s}$:

$$\begin{aligned}
\mathbf{s} &= \mathbf{o} - \alpha\mathbf{d} \\
\mathbf{s_h} &= \mathbf{o_h} - \mathbf{d}\alpha_\mathbf{h} + \alpha - \mathbf{d_h} \\
\alpha &= \frac{\mathbf{n}_S^T\mathbf{o} + d_S}{\mathbf{n}_S^T\mathbf{d}} \\
\alpha_\mathbf{h} &= \frac{(\mathbf{n}_S^T\mathbf{o_h})(\mathbf{n}_S^T\mathbf{d}) - (\mathbf{n}_S^T\mathbf{o} + d_S)(\mathbf{n}_S^T\mathbf{d_h})}{(\mathbf{n}_S^T\mathbf{d})^2}
\end{aligned} \tag{19}$$

The final projection onto the camera image plane $\mathbf{q}$ and subsequent dehomogenisation is given by:

$$\begin{aligned}
\mathbf{q} &= \mathbf{KTs} \\
\mathbf{q_h} &= \mathbf{KTs_h} \\
\widetilde{\mathbf{q}} &= \frac{\begin{pmatrix}\mathbf{q}.x & \mathbf{q}.y\end{pmatrix}^T}{\mathbf{q}.z} \\
\widetilde{\mathbf{q}} &= \frac{\mathbf{q}.z\begin{pmatrix}\mathbf{q}_H.x & \mathbf{q}_H.y\end{pmatrix}^T - \begin{pmatrix}\mathbf{q}.x & \mathbf{q}.y\end{pmatrix}^T\mathbf{q}_H.z}{\mathbf{q}.z^2}
\end{aligned} \tag{20}$$

Where the camera intrinsics $\mathbf{K}$ and extrinsics $\mathbf{T}$ are known. $\mathbf{q}.x$ refers to the x-component of the vector $\mathbf{q}$ (equivalent for y and z)

SCITEPRESS
SCIENCE AND TECHNOLOGY PUBLICATIONS