Optimizing freeform lenses for extended sources with algorithmic differentiable ray tracing and truncated hierarchical B-splines

Heemels, Alexander; Koning, Bart De; Möller, Matthias; Adam, Aurèle

DOI
10.1364/OE.515422

Publication date
2024

Document Version
Final published version

Published in
Optics Express

Research Article Vol. 32, No. 6 / 11 Mar 2024 / Optics Express 9730
Optics EXPRESS

Optimizing freeform lenses for extended sources with algorithmic differentiable ray tracing and truncated hierarchical B-splines

ALEXANDER HEEMELS,[1,*] BART DE KONING,[1,2] MATTHIAS MÖLLER,[2] AND AURÈLE ADAM[1]

[1]Optics Research Group, Faculty of Applied Physics, Delft University of Technology, Lorentzweg 1, Delft, The Netherlands
[2]Numerical Analysis, Faculty of Applied Mathematics, Delft University of Technology, Mekelweg 4, Delft, The Netherlands
[*]a.n.m.heemels@tudelft.nl

Abstract: We propose a method for optimizing the geometry of a freeform lens to redirect the light emitted from an extended source into a desired irradiance distribution. We utilize a gradient-based optimization approach with MITSUBA 3, an algorithmic differentiable non-sequential ray tracer that allows us to obtain the gradients of the freeform surface parameters with respect to the produced irradiance distribution. To prevent the optimizer from getting trapped in local minima, we gradually increase the number of degrees of freedom of the surface by using Truncated Hierarchical B-splines (THB-splines) during optimization. The refinement locations are determined by analyzing the gradients of the surface vertices. We first design a freeform using a collimated beam (zero-etendue source) for a complex target distribution to demonstrate the method's effectiveness. Then, we demonstrate the ability of this approach to create a freeform that can project the light of an extended Lambertian source into a prescribed target distribution.

1. Introduction

Various optical elements can control the light emitted by a source and redirect it into a particular irradiance or intensity distribution. One element, in particular, freeform lenses and reflectors, is of great interest to optical designers as it allows for more flexibility in creating compact illumination systems. However, determining the ideal geometry of a freeform capable of generating a desired irradiance distribution is a complex problem. It has been thoroughly researched, assuming the source is zero-étendue. The problem of finding a suitable freeform geometry can then be formulated as a Monge-Kantorovich mass transport problem, which is then typically solved using the Monge-Ampere equation [1–5], ray mapping methods [6,7], or optimization methods such as the supporting quadratic method [8].

However, the methods are not capable of designing freeforms for extended or finite étendue sources. This issue is often solved by treating it as a perturbation of the zero-étendue problem. This leads to feedback-type methods, which iterate between designing the freeform using zero-etendue design techniques and adjusting the target irradiance or intensity distribution to get closer to the desired irradiance distribution [9–11]. Alternatively, a design can be sought through optimization of the lens geometry directly [12–14].

In this manuscript, we present a gradient-based optimization method capable of designing freeform optics for extended sources without needing the target distribution to be iteratively updated. Furthermore, it avoids local minima by gradually increasing the number of degrees of freedom of the surface. To get the gradients of the lens parameters, we use Algorithmic

#515422 https://doi.org/10.1364/OE.515422
Journal © 2024 Received 8 Dec 2023; revised 30 Jan 2024; accepted 19 Feb 2024; published 4 Mar 2024

Optics EXPRESS

Differentiable ray tracing (AD ray tracing), a new paradigm in lens design that has been successfully applied in imaging [15–18] and illumination applications [19–21]. It offers a simple way to compute the gradient of the parameters describing the optical system without requiring lengthy derivations to obtain a symbolic description or run numerous simulations to calculate the gradients of the lens parameters numerically. It also allows for easier development of new optimization procedures for designing complex optical systems with the outlook of combining the physical simulation in a network training loop [15,16].

In our previous work [19], we demonstrated the effectiveness of non-sequential ray tracing for zero-étendue, yet the proposed method had problems optimizing for finite étendue sources. In addition, we make use of a new ray tracer: Mitsuba 3 [22], an open-source physics-based differentiable ray tracer that performs non-sequential ray tracing with Monte Carlo ray sampling that is GPU compatible and is used to trace the rays from the source to the target and to estimate the radiometric quantities. In non-sequential ray tracing, the path of many rays is traced, and the objects they will interact with are not known beforehand. At every interface of an object the ray interacts with, the ray can be split, meaning that for specular interactions, the ray can be reflected or refracted. Additionally, the Monte Carlo sampling technique randomly selects the initial positions and directions of the rays to obtain an unbiased estimate of the radiometric quantities. This differs from sequential ray tracing, where a limited amount of rays are traced from surface to surface in a predetermined order to eventually calculate the system's aberration coefficients.

B-splines and the nonuniform rational variant NURBS are popular for describing freeform surfaces as they allow local changes to the geometry. However, when creating complex irradiance distributions, freeforms require many degrees of freedom, which can cause the optimizer to get stuck in undesirable local minima.

To address this issue, new degrees of freedom can gradually be added during optimization, such as proposed by Wang et al., [20], who use knot insertion [23]. However, when a knot is added, a decision must be made about whether a whole row or column of the knot span is refined. This adds extra degrees of freedom in areas where none are needed. To refine a single knot span, alternative spline descriptions can be used, such as LR [24], HB [25], THB [26], U [27] and T-splines [28]. T-splines having been proposed in the context of freeform design [29,30]. We make use of *Truncated Hierarchical B-spline* (THB-splines) [26,31].

With these new local refinement possibilities, strategies must be developed to determine where the surfaces should be refined. We present such a method that determines the optimal locations for refinement based on the gradient of the vertices of the surface, which can easily be obtained using algorithm differentiation.

The presented gradient-based optimization scheme can design freeform optics for extended sources and consists of three components: AD ray tracing, THB-splines, and the refinement strategy. AD ray tracing is used to obtain the gradients of the control points of the spline, THB-splines allow local refinement of the surface, and the refinement strategy determines where new degrees of freedom should be added. We first present the systems we are optimizing, followed by an overview of the fundamentals of AD ray tracing, THB-splines, and the optimization procedure. We demonstrate the workings of the method by first designing a freeform for a complex irradiance distribution with a zero-étendue source. Finally, we demonstrate its effectiveness in the design of a freeform using an extended source.

2. Methods

2.1. System definition

The system under consideration is depicted in Fig. 1(a); it consists of a freeform lens with a refractive index of n, a source, and a target plane where the irradiance is measured; the overview of the parameters used to fully define the system are depicted in Fig. 1(b). The source can be

Optics EXPRESS

either a collimated bundle of light or an extended rectangular source with Lambertian emission with its centroid on the optical axis and a normal perpendicular to it. The distance from the source to the origin of the first lens surface (O_{front}) is given by d_0, the distance between the origins of the front and the rear surface (O_{rear}) by d_1, and the distance between the origin of the rear surface and the target plane by d_2. The surfaces of the lens are described in terms of local coordinates (u, v) and consist of a base conic and a sag given by a B-spline:

$$z_{\text{surface}}(u, v) = z_{\text{conic}}(R, K, u, v) + z_{\text{B-spline}}(\mathbf{C}, u, v). \tag{1}$$

The base conic z_{conic} depends on its radius of curvature R and conic constant K. A B-spline surface is defined using B-spline basis functions $N_{i,p}(u)$ and $N_{j,q}(v)$ which require a specified degree p and q and knot vectors $U = \{u_0, \ldots, u_{N_u-1}\}$ and $V = \{v_0, \ldots, v_{N_v-1}\}$ with N_u and N_v denoting the number of knots in the u and v directions. The Cox-The Boor relations [23, Eq. (2.5)] are used to calculate the B-spline basis functions which span the spline spaces $\mathcal{U}$ and $\mathcal{V}$.

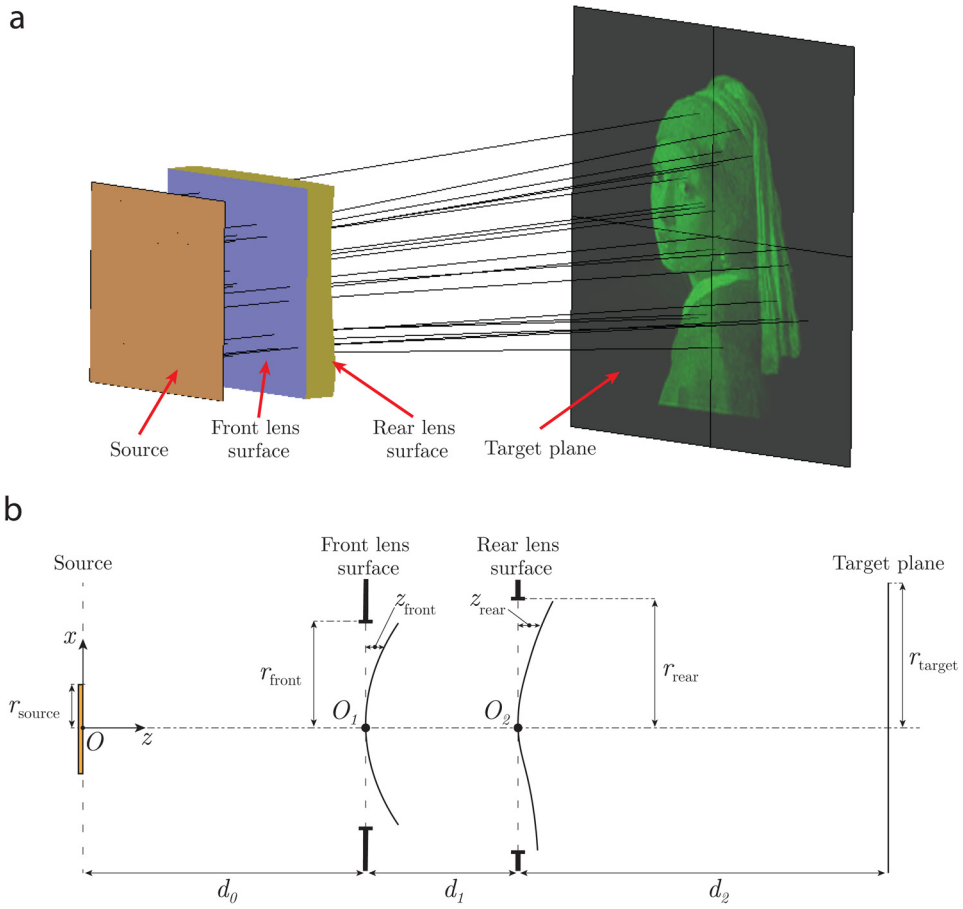

Fig. 1. (a) 3D model of the system consisting of a source, freeform lens, and a target plane where the irradiance is measured; (b) Simplified, 2D view of the system depicting the different parameters used to define the system.

To construct the B-spline surface, the splines control points $\mathbf{c}_{i,j,k} = \begin{bmatrix} c^x_{i,j,k} & c^y_{i,j,k} & c^z_{i,j,k} \end{bmatrix}^T$ have to be specified, with $i \in \{0, \ldots, N_u + p - 1\}$ and $j \in \{0, \ldots, N_v + q - 1\}$ and $k \in \{\text{front}, \text{rear}\}$. By

Optics EXPRESS

collecting B-spline basis functions in a vector $\mathbf{N}_p(u) = \begin{bmatrix} N_{0,p}(u) & N_{1,p}(u) & \dots & N_{N_u+p-1,p}(u) \end{bmatrix}^T$ we can write B-spline surface as a tensor product [23, Eq. (3.11)]:

$$z_{\text{B-spline}}(u, v) = \mathbf{N}_p(u)^T \mathbf{C} \mathbf{N}_q(v), \tag{2}$$

with

$$\mathbf{C} = \begin{bmatrix} \mathbf{c}_{0,0} & \mathbf{c}_{0,1} & \dots & \mathbf{c}_{0,N_u+p-1} \\ \mathbf{c}_{1,0} & \mathbf{c}_{1,1} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{c}_{N_v+q-1,0} & \mathbf{c}_{N_v+q-1,1} & \dots & \mathbf{c}_{N_v+q-1,N_u+p-1} \end{bmatrix}. \tag{3}$$

In this manuscript, without loss of generality, the B-spline surfaces are of degree p in both u and v direction, and therefore, the degree of the spline will be omitted from the notation from now on. The whole system can then be described using the set of parameters

$$\boldsymbol{\pi} = \{r_{\text{source}}, r_{\text{front}}, r_{\text{rear}}, r_{\text{target}}, d_0, d_1, d_2, n, R_{\text{front}}, K_{\text{front}}, \mathbf{C}_{\text{front}}, p_{\text{front}}, R_{\text{rear}}, K_{\text{rear}}, \mathbf{C}_{\text{rear}}, p_{\text{rear}}\}. \tag{4}$$

2.2. Differentiable ray tracing background

To determine the irradiance at the target plane, we first calculate the trajectory of N_k rays emitted by a planar source with a radiance L_{source} at a point (x, y) on its surface in a direction (θ, ϕ). These spatial and angular coordinate pairs are combined into a 4D ray coordinate $\mathbf{x} = (x, y, \theta, \phi)$. After propagating through the system N'_k rays end up on the target screen with a radiance value L_{target} at the target ray coordinate $\mathbf{x}' = (x', y', \theta', \phi')$. The irradiance values are then obtained by discretizing the target plane by placing a grid of $N_t \times M_t$ pixels at the points $P_{n,m} = \{(2n/N_t - 1)r_{\text{target}}, (2m/M_t - 1)r_{\text{target}}\}$, where $n \in \{0, \dots, N_t\}$ and $m \in \{0, \dots, M_t\}$. The value of each pixel is determined by summing the radiance values of the rays intersecting the target screen in an area $\mathcal{A}$ around the pixel. The pixel measurement function or reconstruction filter $W(\mathbf{x}')$ determines this area and the weight attributed to a ray. Common choices are box- and Gaussian filters, depicted in Fig. 2(a1 and a2). The pixel's value is then determined by weighting all the rays to the pixel's measurement function as shown in Fig. 2(b1 and b2) and summing them together as shown in Fig. 2(c1 and c2)

$$E_{n,m}(\boldsymbol{\pi}) = \sum_{k \in \mathcal{A}_{n,m}} L(x'_k, y'_k, \boldsymbol{\pi}) W_{n,m}(x'_k, y'_k). \tag{5}$$

All irradiance values are then combined to give the irradiance matrix $\mathbf{E}(\boldsymbol{\pi}) \in \mathbb{R}^{N_t \times M_t}$. To achieve the desired irradiance distribution $\mathbf{E}_{\text{desired}}$, we need to optimize the system parameters $\boldsymbol{\pi}$. This is done by comparing the irradiance distribution produced by the system $\mathbf{E}$ with the desired one with a metric $\mathcal{L}$. To determine how the system parameters should be changed, we calculate the gradient of $\mathcal{L}$ with respect to a parameter π_k. This gradient is obtained by applying the chain rule, resulting in the following gradient for the n, m-th irradiance value:

$$\frac{\partial}{\partial \pi_k} \mathcal{L}\left(\mathbf{E}(\pi_k)\right) = \sum_{n=1}^{N_t} \sum_{m=1}^{M_t} \frac{\partial \mathcal{L}\left(\mathbf{E}(\pi_k)\right)}{\partial E_{n,m}} \frac{\partial E_{n,m}}{\partial \pi_k}. \tag{6}$$

Gradients can be evaluated in different ways, e.g., using numerical or algorithmic differentiation. To evaluate the derivative of Eq. (7) using numerical differentiation [32, Chapter 8.1], we can use

Optics EXPRESS

central differences to obtain a second-order accurate approximation

$$\frac{\partial \mathcal{L}}{\partial \pi_i}(\boldsymbol{\pi}) = \frac{\mathcal{L}(\boldsymbol{\pi} + \varepsilon \mathbf{e}_i) - \mathcal{L}(\boldsymbol{\pi})}{\varepsilon} + O(\varepsilon^2). \tag{7}$$

A total of $n+1$ function evaluations are required to evaluate Eq. (5) when a total of n parameters are used. This becomes unreasonable for a large number of degrees of freedom; for instance, calculating the gradient for a lens with 10×10 control points would require 101 ray tracing simulations. Alternatively, *algorithmic differentiation* can be used. It is based on two core concepts. First, most functions can be expressed using simple operations of one or two variables, such as addition, multiplication, logarithms, powers, or geometric identities. The second is the chain rule, which states that if h is a function of a vector $\mathbf{y} \in \mathbb{R}^m$, which itself is a function of $\mathbf{x} \in \mathbb{R}^n$ we can calculate the derivative of h with respect to $\mathbf{x}$ as:

$$\nabla h(\mathbf{y}(\mathbf{x})) = \sum_{i=1}^{m} \frac{\partial h}{\partial y_i} \nabla y_i(\mathbf{x}). \tag{8}$$

Unlike numerical differentiation, algorithmic differentiation only requires a ray tracing simulation, during which a computational graph is constructed, and is used to evaluate the gradient. For a detailed overview of the topic of algorithmic differentiation, the reader is referred to [33] and [32, Chapter 8.2]. The particular combination of AD with ray tracing is discussed in more detail in [22,34].

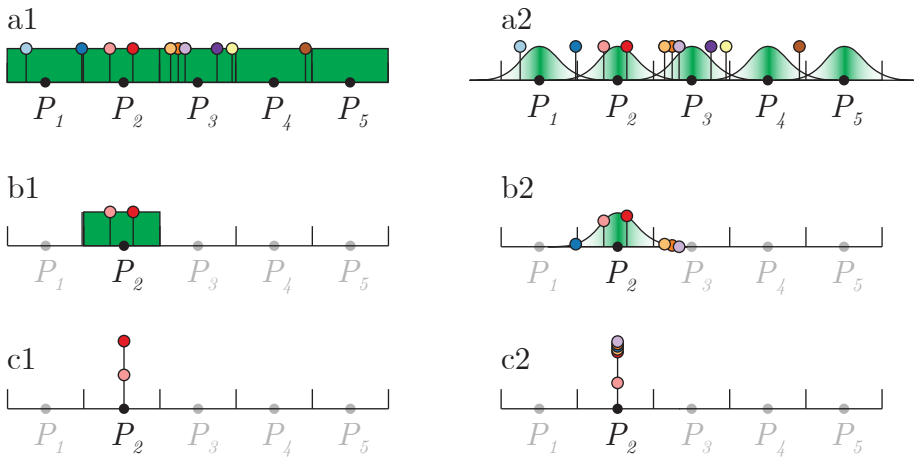

Fig. 2. Process of determining pixel values for a box filter and Gaussian filter: (a1 - a2) pixels with corresponding measurement functions and rays; (b1 - b2) rays contributing to pixel P_2 with corresponding measurement function; (c1 - c2) final pixel weight obtained by adding ray contributions.

2.3. B-splines refinement and truncated hierarchical B-splines (THB-splines)

To increase the degrees of freedom in a B-spline, knot insertion or refinement can be used [23, Chapter 5.2 and 5.3]. These operations add one or multiple knots into the knot vector, enabling the addition of new control points. These refinement procedures are possible because the B-spline basis functions are refinable, meaning that they can be written as a linear combination of basis functions with smaller support. To refine a B-spline surface through knot insertion, a *refinement matrix* $\mathbf{R}_u \in \mathbb{R}^{N_u+p-1 \times N_u+p}$ and $\mathbf{R}_v \in \mathbb{R}^{N_v+p-1 \times N_v+p}$ are constructed, which relate the

Optics EXPRESS

B-spline basis functions with smaller support $\widetilde{\mathbf{N}}(u) \in \mathbb{R}^{N_u+p}$ and $\widetilde{\mathbf{N}}(v) \in \mathbb{R}^{N_v+p}$ to the original basis functions $\mathbf{N}(u) \in \mathbb{R}^{N_u+p-1}$ and $\mathbf{N}(v) \in \mathbb{R}^{N_v+p-1}$:

$$\mathbf{N}(u) = \mathbf{R}_u \widetilde{\mathbf{N}}(u), \ \mathbf{N}(v) = \mathbf{R}_v \widetilde{\mathbf{N}}(v) \tag{9}$$

The refined surface with the new B-spline basis can then be written as

$$z_{\text{B-spline}}(u, v) = \widetilde{\mathbf{N}}(u)^T \widetilde{\mathbf{C}} \widetilde{\mathbf{N}}(v), \tag{10}$$

where $\widetilde{\mathbf{C}} = \mathbf{R}_u^T \mathbf{C} \mathbf{R}_v$. Inserting a knot into the knot vector V will add a control point in the v direction, except when a knot is inserted at a position where a knot already exists. However, this introduces $N_u + p - 1$ control points in the u direction and can lead to over-refinement of the B-spline surface. To illustrate this problem, let us consider a B-spline surface of bi-degree 3 with knot vectors $U = \{0, 0, 0, 0.25, 0.5, 0.75, 1, 1, 1\}$ and $V = \{0, 0, 0, 0.25, 0.5, 0.75, 1, 1, 1\}$. The dimension of the B-spline basis vectors $\mathbf{N}(u)$ and $\mathbf{N}(v)$ are both 6. Therefore, the number of control points required is 36. The knot spans generated by the knot vectors U and V are shown in Fig. 3. Suppose we want to refine the elements $[0, 0.25) \times [0, 0.25)$ and $[0.25, 0.5) \times [0.5, 0.75)$, which are highlighted in red in Fig. 3(a-c). One way to achieve this refinement is by inserting a knot $\widetilde{u} = 0.4$ into the knot vector U, which increases the dimension of $\mathbf{N}(u)$ to 7, and thus the number of control points becomes 42. Subsequently, a knot $\widetilde{v} = 0.1$ can be inserted into the knot vector V, raising the number of required control points to 49. While we only wanted to refine two knot spans, seven have been refined, adding 13 degrees of freedom. In addition, as knot insertion refines along rows or columns, there are numerous ways in which the two knot spans can be refined. This issue is solved by using HB-splines [25] and THB-splines [26,31], which allow a single knot span to be refined by adjusting the basis functions, which are not contained in the knot spans which are to be refined.

To construct truncated hierarchical B-splines a sequence of nested spline spaces is required of levels $\ell \in \{0, 1, \ldots, L\}$, $\mathcal{V}^0 \subset \mathcal{V}^1 \subset \ldots \subset \mathcal{V}^L$ defined on a domain $\Omega_v^0 \in [0, 1]$. Starting with a knot vector of level $\ell = 0$, the knot vectors generating these spline spaces of any level ℓ

$$V^\ell = \left\{ v_0^\ell, \ldots, v_m^\ell \right\}, \tag{11}$$

can be obtained by bisecting the knot vector of level ℓ to obtain the knot vector of level $\ell + 1$

$$V^{\ell+1} = \left\{ v_0^{\ell+1} = v_0^\ell, v_1^{\ell+1} = \frac{v_0^\ell + v_1^\ell}{2}, v_2^{\ell+1} = v_1^\ell, \ldots, v_{2m-2}^{\ell+1} = v_{m-1}^\ell, v_{2m-1}^{\ell+1} = \frac{v_{m-1}^\ell + v_m^\ell}{2}, v_{2m}^{\ell+1} = v_m^\ell \right\}. \tag{12}$$

We denote the vector of B-spline basis functions of level ℓ as $\mathbf{N}^\ell$ and using knot insertion (Eq. (9)) we can relate the B-spline basis vector of levels ℓ and $\ell + 1$ using a refinement matrix $\mathbf{R}_u^{\ell+1} \in \mathbb{R}^{2^\ell (N_u+p-1) \times 2^{\ell+1}(N_u+p-1)}$

$$\mathbf{N}^\ell(u) = \mathbf{R}_u^{\ell+1} \mathbf{N}^{\ell+1}(u). \tag{13}$$

To indicate where on the lens domain $\Omega^0 = \Omega_u^0 \times \Omega_v^0$ the higher levels should be, we use nested domains $\Omega^0 \supseteq \Omega^1 \supseteq \ldots \supseteq \Omega^L$. In Figs. 4(a2-c2), an example nested domain is shown, with the lens domain Ω_0 shown in Fig. 4(a2), together with the knot lines, indicating where in Ω_u and Ω_v the knots are located. We get the lens' hierarchical mesh by overlaying all these domains, as shown in Fig. 4(a1). We define a characteristic matrix $\mathbf{X}_u^\ell$ and $\mathbf{X}_v^\ell$ of $\mathbf{N}(u)^\ell$ and $\mathbf{N}(u)^\ell$ which are used to relate Ω^ℓ with $\Omega^{\ell+1}$ as follows:

$$\mathbf{X}_u^\ell = \text{diag}(x_{u,i}^\ell), \quad x_{u,i}^\ell = \begin{cases} 1, \text{ if supp } N(u)_i^\ell \subseteq \Omega_u^\ell \text{ and supp } N(u)_i^\ell \nsubseteq \Omega_u^{\ell+1}, \\ 0, \text{ otherwise.} \end{cases} \tag{14}$$

The elements of the characteristic matrix are set to one when the support of a B-spline basis function is fully contained within a domain Ω^ℓ and does not fall within $\Omega^{\ell+1}$. Otherwise, it is set

Optics EXPRESS

to zero. Then, we can define the truncation matrix [31] as

$$\mathbf{T}_u^\ell = \mathbf{R}_u^\ell(\mathbf{I}^\ell - \mathbf{X}_u^\ell), \tag{15}$$

with $\mathbf{I}$ the identity matrix. Using Eq. (15) we can recursively determine the THB-coefficient matrix $\mathbf{D}^\ell$

$$\mathbf{D}^\ell = \mathbf{T}_u^{\ell T}\mathbf{D}^{\ell-1}\mathbf{T}_v^\ell + \mathbf{X}_u^{\ell T}\mathbf{C}^\ell\mathbf{X}_v^\ell, \tag{16}$$

starting with $\mathbf{D}^0 = \mathbf{C}$. Finally, a THB-spline with highest refinement level L can then be expressed in terms of $\mathbf{N}^L$:

$$z_{\text{THB-spline}}(u, v) = \mathbf{N}^L(u)^T\mathbf{D}^L\mathbf{N}^L(v). \tag{17}$$

In Fig. 4(b1) an THB-spline of the hierarchical mesh in Fig. 4(a1). It shows how areas with higher refinement levels can incorporate finer details into the surface. For a more detailed discussion on the construction of the THB-basis functions, the reader is referred to Section 2 of the Supplement 1.

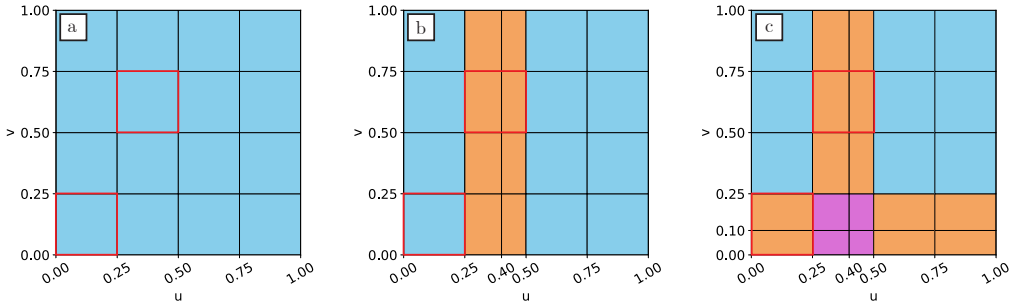

Fig. 3. Knot spans of the B-spline surface at different refinement steps, the areas which should be refined are highlighted in red, orange areas indicate the knot spans which have been refined once, the purple areas indicate the knot spans which have been refined twice: (a) Original knot span; (b) Knot span after the knot $\widetilde{u} = 0.4$ is inserted; (c) Knot span after the knot $\widetilde{v} = 0.1$ is inserted;

2.4. Optimization procedure

At the beginning of the optimization, the B-spline surfaces control points $\mathbf{c}_{i,j,k}$ are uniformly distributed over the surface. Of these control points, the x and y coordinate $c^x_{i,j,k}$ and $c^y_{i,j,k}$ are fixed, while only $c^z_{i,j,k}$ is variable. This ensures that the control points cannot move into positions such that the B-spline surface would intersect itself. To limit the maximum deviation, we choose to optimize a weight $w^{i,j,k} \in \mathbb{R}$, with the set of all weight denoted by $\mathcal{W} = \{w^{i,j,k}\}$. These weights are directly related to the z-postion of the control points by:

$$c_z^{i,j,k} = \arctan\left(\frac{w^{i,j,k}}{z_{\text{freedom}}}\right). \tag{18}$$

Using the arctangent, we can restrict the maximum deviation of $c^z_{i,j,k}$ to z_{freedom}. These weights are then used to minimize the *Frobenius norm* of the difference between the irradiance measured

Optics EXPRESS

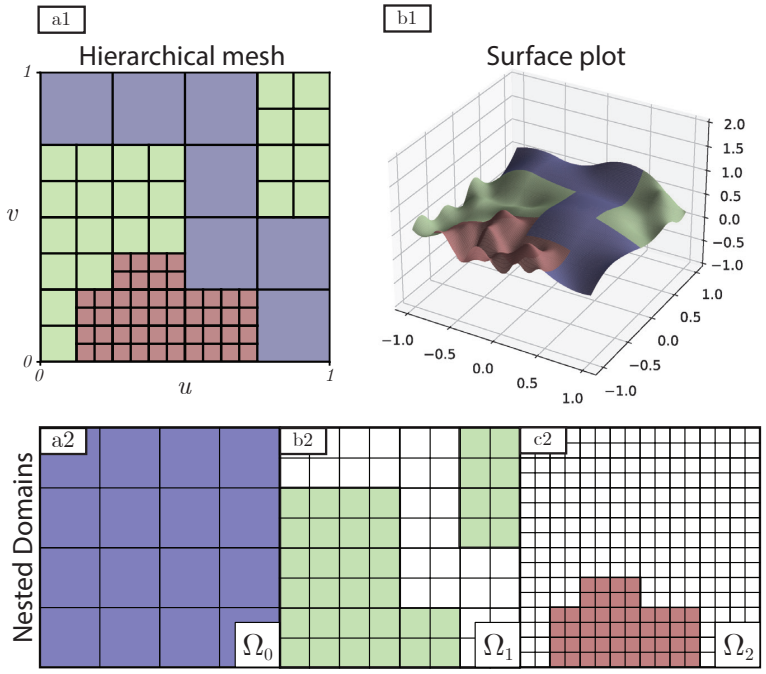

Fig. 4. (a1) Hierarchical mesh; (b1) surface plot of the THB-spline with randomly z-positions of the control points; (a2-c2) the nested domains $\Omega_0, \Omega_1, \Omega_2$ with the colored areas indicating where the nested domains are active.

at the target plane $\mathbf{E}^{\mathcal{W}}_{\text{target}}$ which is a variable of the weights $\mathcal{W}$ and the desired irradiance $\mathbf{E}_{\text{desired}}$:

$$\mathcal{L}(\mathcal{W}) = \sqrt{\sum_{n=1}^{N}\sum_{m=1}^{M}\left|\mathbf{E}^{\mathcal{W}}_{\text{target}}[n,m] - \mathbf{E}_{\text{desired}}[n,m]\right|^2}. \tag{19}$$

To minimize Eq. (19), we utilize a gradient-based optimization approach by making use of the PyTorch toolbox [35] and the ADAM optimizer [36]. The gradient with respect to the weight $\mathcal{W}$ is obtained using algorithmic differentiable ray tracing.

We start the optimization by initializing the surface with a low number of control points and perform optimization for a fixed set of iterations. Once finished, we identify areas where the surface requires refinement by examining how changes in the vertex positions affect the irradiance distribution. These vertices are located on a grid G withing the domain $\mathbf{\Omega} = [0,1]^2$ with the x and y positions of the points:

$$G_{i,j} = \left(\frac{i}{N_g - 1}, \frac{j}{M_g - 1}\right) \quad \text{with} \quad i,j \in \mathbb{N},\ 0 \le i \le N_g - 1,\ 0 \le j \le M_g - 1. \tag{20}$$

These vertices fall with knot span products, which are domains between adjacent knots represented by $C^{\ell}_{r,s} = [u^l_{r,s}, u^l_{r+1,s}) \times [v^l_{r,s}, v^l_{r,s+1})$ which subdivided the surface. Each knot span product is identified by its index rs and its refinement level ℓ. To determine which knot spans need refinement, we calculate the gradients of the z-positions of the vertices, denoted as $Z_{i,j}$, within a knot span product and sum the absolute values, given by

$$\mathcal{Z}^{\ell}_{r,s} = \sum_{i,j \in C^{\ell}_{r,s}} \left|\frac{\partial \mathcal{L}(Z_{i,j})}{\partial Z_{i,j}}\right|. \tag{21}$$

Optics EXPRESS

If the absolute sum of the vertex gradients with a knot span exceeds the average absolute gradient by a threshold value α:

$$\mathcal{Z}^{\ell}_{r,s} > \frac{\alpha}{2^{\ell}(N_u + p - 1)^2} \sum_{r',s'} \mathcal{Z}^{\ell}_{r',s'}, \tag{22}$$

the cell is refined, and a new optimization batch is started. We repeat this process of optimization and refinement until the finest refinement level L of the THB-surface is reached.

Section 3 of the Supplement 1 demonstrates the method by fitting a THB-spline to a height map.

3. Results

We demonstrate the working of the proposed optimization method with two design examples. The first design is a freeform, illuminated by a collimated beam of light that projects the image of the "Girl with the Pearl Earring." The second design projects a smooth, curved irradiance distribution with an extended source close to the front surface. All simulations were done on a desktop computer with an Intel Xeon CPU E5-1620 v3 and NVIDIA RTX 2080 Ti GPU.

A Gaussian measurement function is used to ensure the stability of the gradient calculations of the surface control points. This is because the Gaussian measurement functions of neighboring pixels slightly overlap, preventing rays from suddenly jumping from one pixel to another. In addition, the standard deviation of the Gaussian was tuned to smooth out the vertex gradients. This was necessary as small standard deviations produced noisy gradients, making the refinement procedure unstable. MITSUBA 3 does not provide a direct method of measuring irradiance values for a grid of pixels. Therefore, the light redirected by the freeform is projected onto the target screen and imaged by a pinhole camera, giving an unaberrated image of the irradiance distribution used in the optimization. More details are found in the Section 4 of the Supplement 1.

It is also possible to design freeforms with B-splines with a static number of degrees of freedom. In Section 1 of the Supplement 1 we compare results for a simple top-hat distribution and the image of the girl with the pearl earring with B-splines with various static numbers of degrees of freedom. We learn from these examples that large degrees of freedom with large step sizes lead to highly irregular lenses. This can be mitigated to some extent by choosing a smaller step size, leading to a longer optimization time, thus, giving a trade-off between the number of degrees of freedom and the total optimization time. In addition, for the zero-etendue example, we compare THB-splines with regular B-spline refinement, where all knot spans are refined after a fixed number of iterations. All results are compared to LightTools [37], in which the system is recreated and simulated. A detailed description of how the systems are modeled in both MITUSBA and LightTools is presented in Section 4 of the Supplement 1, and the lens before and after optimization and the corresponding LightTools models are shown in Section 5 of the Supplement 1. Finally, we look at the size of the knot spans as their size indicates whether or not diffraction due to small details on the lens could be problematic [38].

3.1. THB-spline freeform lens optimization zero-étendue

We optimize the rear surface of the freeform lens, which is illuminated by a uniform collimated beam of light, with all rays having the same radiance, to generate the girl with the pearl earring (Fig. 5(a1)) at a distance of 300 mm of the lens. A radius of 100 mm and a conic constant of -1 were chosen for the base conic surface such that the initial irradiance distribution was slightly smaller than the desired target distribution. The spline surface was initialized as a B-spline of degree 3, and a control net of 10×10 and z_{freedom} was chosen to be a quarter of the thickness of the lens. The values of all other system parameters can be found in Table 1.

Figure 5(b1) shows the chosen learning rates and refinement parameters used during optimization batches. The learning rates were selected to minimize the number of iterations required

Optics EXPRESS

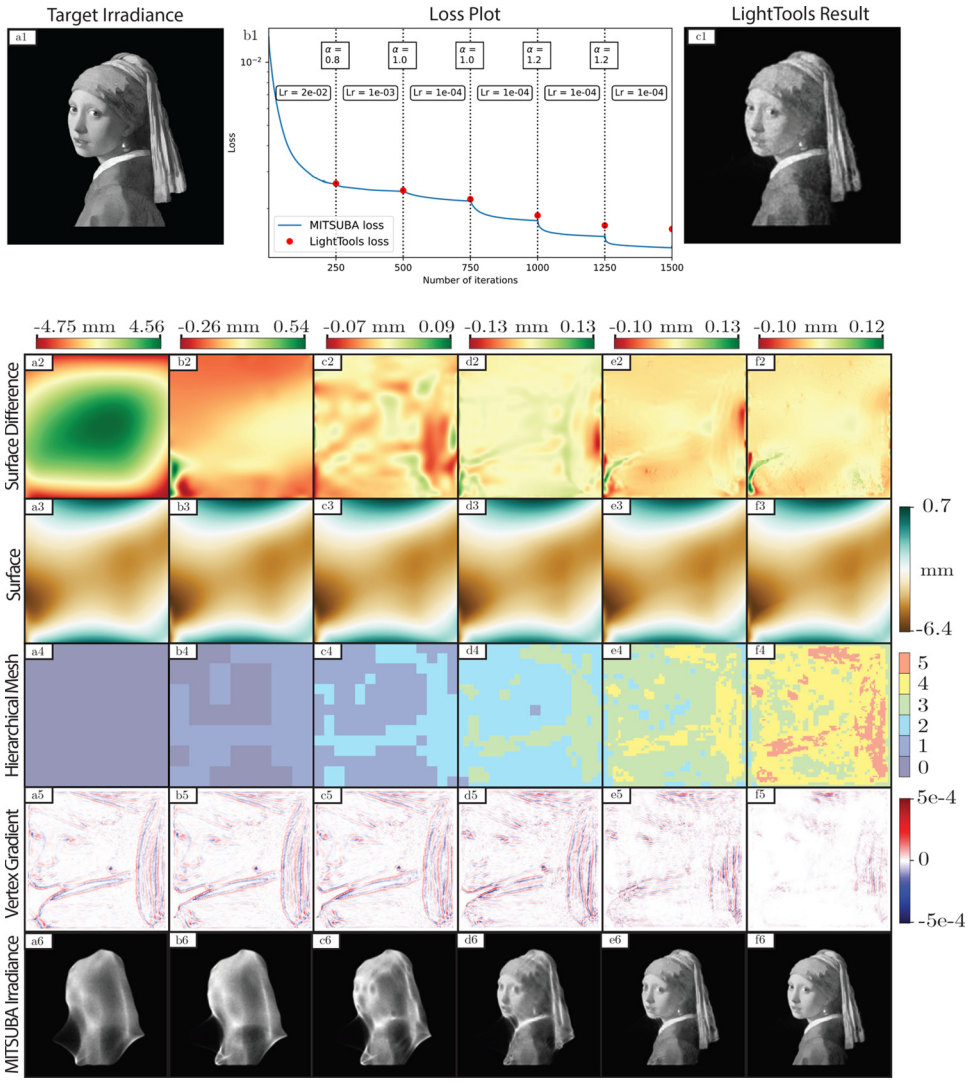

Fig. 5. Results using THB-refinement strategy (a1) Desired irradiance distribution; (b1) Plot of loss through optimization with refinement parameters and learning rates and LightTools loss at the end of each batch; (c1) Irradiance obtained from LightTools; (a2-f2) The difference in the surface between subsequent optimization steps; (a3-f3) Surface at the end of each optimization step; (a4-f4) Hierarchical mesh of the THB-spline, showing which knot span products are refined; (a5-f5) Gradients of the vertices of the surface; (a6-f6) Obtained irradiance after every optimization batch.

Table 1. System parameters for collimated by light with girl with the pearl earring as target distribution

System Parameters				Surface parameters			
r_{source}	47.5 mm	d_0	N/A mm	R_{front}	∞ mm	N_u, M_v	14
r_{front}	50 mm	d_1	20 mm	K_{front}	0	p, q	3
r_{rear}	50 mm	d_2	300 mm	R_{rear}	100 mm	N_g, M_g	449
r_{target}	100 mm	N_t, M_t	256	K_{rear}	-1	L	5
W	Gaussian	σ_W	1	z_{freedom}	5 mm	n	1.5

Optics EXPRESS

for the optimization to converge and to be small enough to avoid instability. Furthermore, the refinement parameters were chosen first to refine the whole lens surface and then prioritize areas of high vertex gradients in later stages to correct the fine details in the desired irradiance distribution. It took 7 minutes and 31 seconds to optimize the lens. When analyzing the changes in the irradiance distribution after each optimization batch, as expected, the rough contour is shaped. With increasing refinement level of the THB-spline, finer details are filled in in the irradiance. However, the first two refinement steps do not significantly reduce the total loss, as seen in the minor changes in the irradiance distributions in (Figs. 5(b6 and c6)) compared to the initial optimization batch. Examining the vertex gradients, in which we can recognize a distorted version of the target distribution, reveals that vertex gradients remain mostly unchanged until the third refinement step Figs. 5(a5-c5). This is because the high vertex gradients vary with a small area, requiring a high refinement level to eliminate them. As can be seen in the hierarchical meshes shown in Figs. 5(a4-f4), it is precisely around these areas that the THB-spline has the highest refinement level.

One area where the optimizer has an issue is the dark area between the chin and the dress. Initially, it attempts to move the light off the target plane to prevent it from contributing to the irradiance distribution. However, upon close examination, we see that the size of this area is reduced, as in the distorted image which can be seen in the vertex gradients Figs. 5(a4-e4), the chin and the dress move closer toward each other, causing the size of the area which discards light to decrease.

Fig. 6 shows the results for the B-spline. No significant differences can be noticed either in the surface or irradiance distributions. Comparing the loss values shown in Fig. 7, we see they both end up at comparable loss values. However, we observe that the THB-spline results were obtained using fewer degrees of freedom than the B-spline Table 2.

Table 2. Numbers of degrees of freedom at the end of each optimization batch for both B- and THB-splines

	250	500	750	1000	1250	1500
THB-spline	100	161	387	1297	3818	11955
B-spline	100	289	961	3481	13225	51529

We can see that the results obtained in MITUSBA 3 (Fig. 5(f6)) and the verification in LightTools (Fig. 5(c1)) are visually consistent with each other. The way the LightTools loss decreases compared with the MITUSBA loss is shown in Fig. 5(b1) and Fig. 6(b1). For the first three batches, the LightTools loss matches the MITSUBA loss. However, for the final three batches, the difference in the loss increases. As this difference increases during the final batches, the mismatch is likely caused by errors introduced by importing the THB surface into LightTools.

Of the total power emitted by the source (1 Watt), 0.92 Watts end up in the final distribution. At the highest refinement, a knot span has a size of 156×156 μm^2. As we work with degree-3 splines, the smallest change in the lens surface has a support of 468×468 μm^2, which is roughly one thousand times larger than the wavelength size. This allows us to disregard diffraction effects.

3.2. THB-spline freeform lens optimization finite-étendue

To demonstrate the effectiveness for an extended source, a smooth curved target distribution, depicted in Fig. 8(a1). The reasoning behind the simpler target compared to the zero-etendue example is that the use of a finite étendue source limits the details that can be achieved [39–41]. The basic conics of the front and rear surfaces are set such that the initial irradiance distribution was of similar size to the desired distribution, and the B-spline of the rear surface was initialized with 5×5 control points while the front surface is left unchanged throughout the optimization

Optics EXPRESS

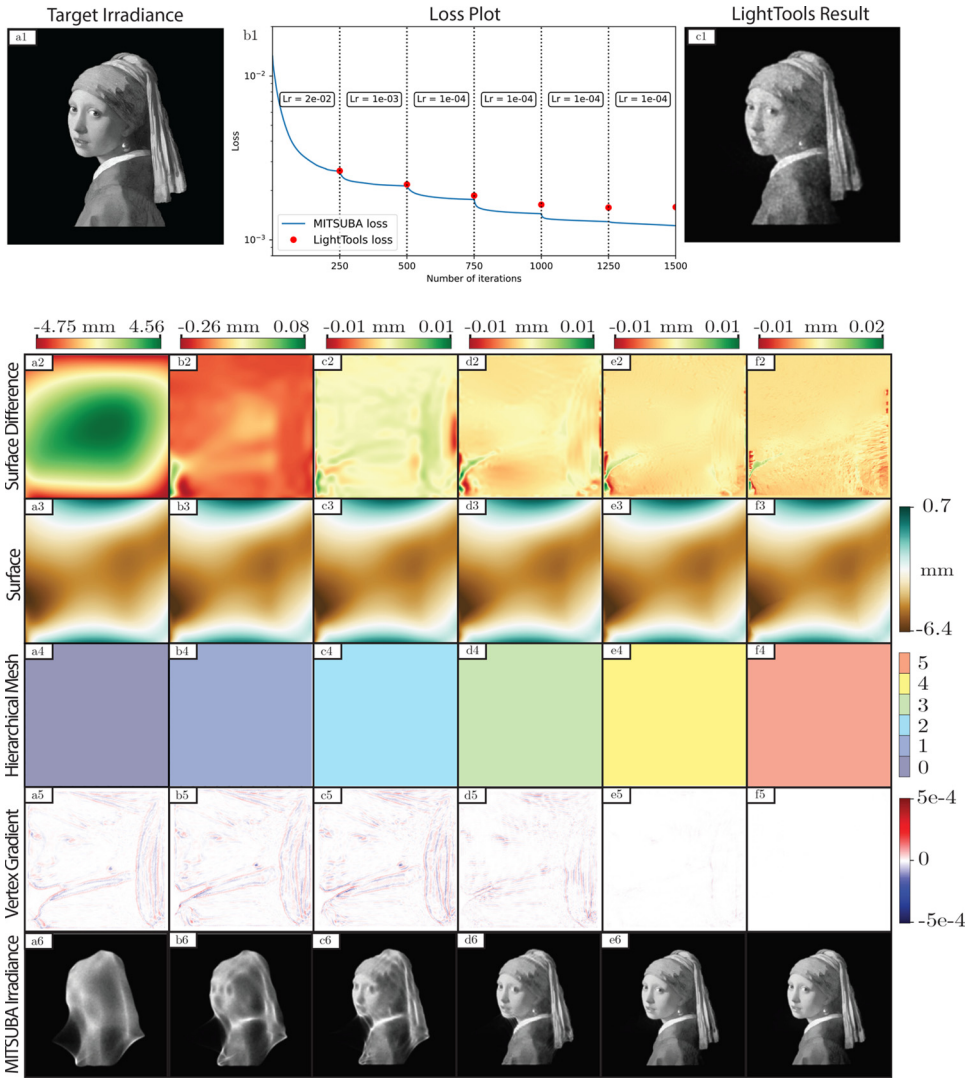

Fig. 6. Results by using B-spline refinement strategy (a1) Desired irradiance distribution; (b1) Plot of loss through optimization with refinement parameters and learning rates and LightTools loss at the end of each batch; (c1) Irradiance obtained from LightTools; (a2-f2) The difference in the surface between subsequent optimization steps; (a3-f3) Surface at the end of each optimization step; (a4-f4) Hierarchical mesh of the THB-spline, showing which knot span products are refined; (a5-f5) Gradients of the vertices of the surface; (a6-f6) Obtained irradiance after every optimization batch.

Optics EXPRESS

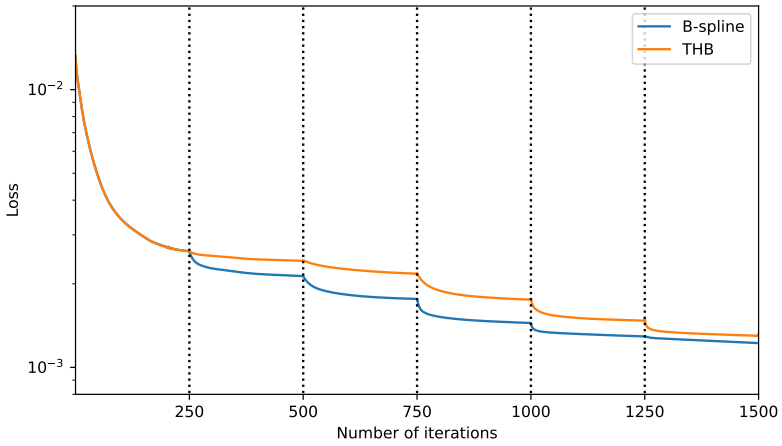

Fig. 7. Loss comparison between THB and B-spline.

and z_{freedom} was chosen to be roughly a quarter of the thickness of the lens. An overview of all the system parameters can be found in Table 3.

Table 3. System parameters: extended source with curved uniform target distribution

System parameters				Surface parameters			
r_{source}	0.5 mm	d_0	2 mm	R_{front}	−5 mm	N_u, M_v	9
r_{front}	5 mm	d_1	5 mm	K_{front}	-1	p, q	3
r_{rear}	8 mm	d_2	1000 mm	R_{rear}	-10 mm	N_g, M_g	450
r_{target}	2000 mm	N_t, M_t	256	K_{rear}	-1	L	5
W	Gaussian	σ_W	1	z_{freedom}	1.5 mm	n	1.5

The learning rates and refinement parameters are shown in Fig. 8(b1). The learning rates were again chosen to balance the number of iterations to the stability of the optimization. The refinement parameters were chosen to ensure that the mirror symmetry of the hierarchical mesh was maintained as long as possible throughout the refinement process. The preservation of symmetry was critical in the earlier stages, as breaking it would result in different areas of the lens being optimized at different refinement levels, leading to worse-performing lenses. It took 29 minutes and 23 seconds to optimize the lens. At the end of each batch, the number of control points is 25, 49, 70, 253, 637, and 2180.

The graph depicting the loss shows that each refinement significantly reduces the loss. Why this is the case can be understood by analyzing the vertex gradients. In Figs. 8(a5-d5), we can identify two distinct gradient areas: a large area in the center with a structured pattern and a ring surrounding it. It is important to note that although these areas seem disconnected, they are connected. This is due to the chosen standard deviation of the Gaussian measurement function, which causes the gradients in this intermediate area to be much smaller than in the ring or central parts of the vertex gradients when blurred. During the first four optimization batches, the structured details in the central area gradually decrease in size (Figs. 8(a5-d5)), shaping the central, uniform part of the irradiance distribution (Figs. 8(a6-d6)). However, after the fifth optimization batch, unstructured and low-valued gradients dominate the central domain, while the edges show large gradients, as seen in Fig. 8(e4). In the final optimization batch, the area of the rear surface, which corresponds to the edges in the vertex gradients, is used to correct the

Optics EXPRESS

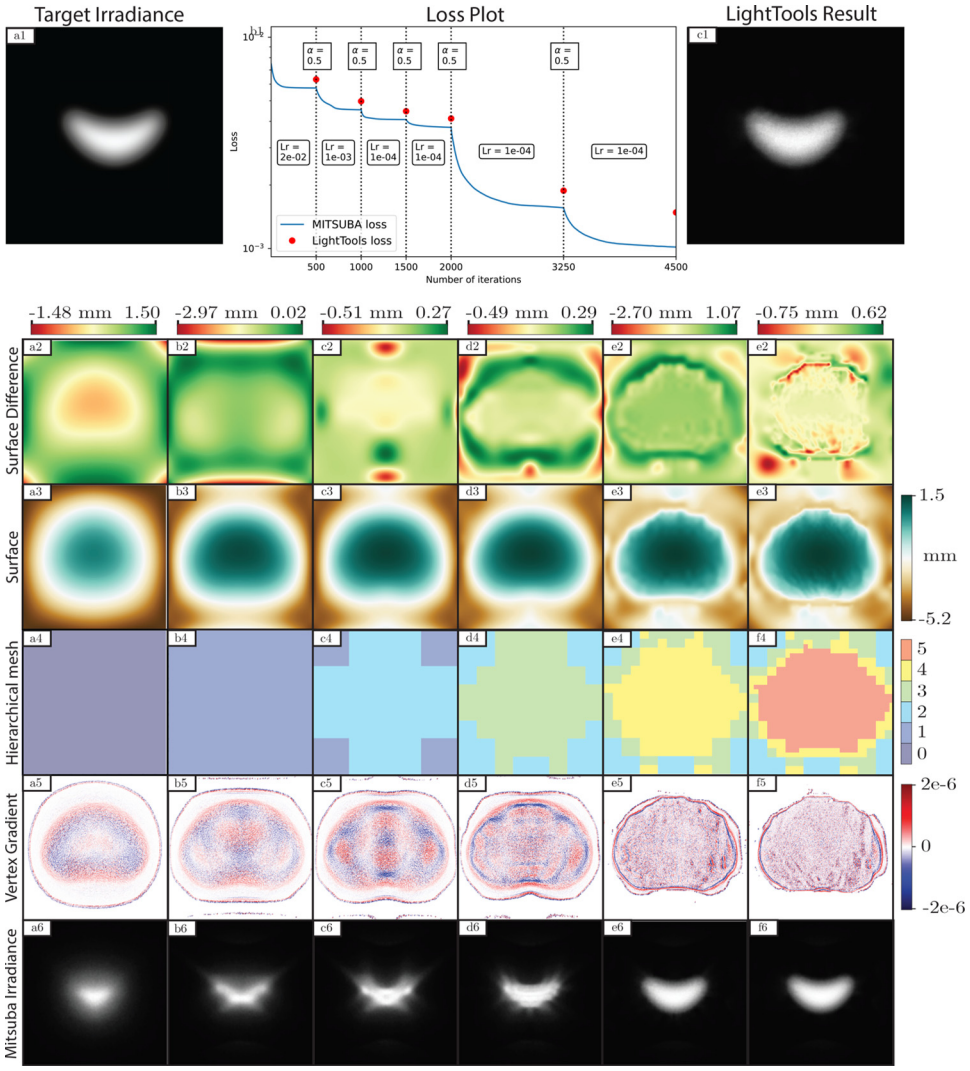

Fig. 8. (a1) Desired irradiance distribution; (b1) Plot of loss through optimization with refinement parameters and learning rates; (c1) Irradiance obtained from LightTools; (a2-f2) The difference in the surface between subsequent optimization steps; (a3-f3) Surface at the end of each optimization step; (a4-f4) Hierarchical mesh of the THB-spline, showing which knot span products are refined; (a5-f5) Gradients of the vertices of the surface; (a6-f6) Obtained irradiance after every optimization batch.

Optics EXPRESS

thin lines of light protruding from the center of the distribution Fig. 8(e6). These corrections can also be seen when looking at the changes made to the freeform surface during optimization, Fig. 8(f1). The central area is largely left unchanged. Thus, the lower THB levels fill the uniform distribution, while the distribution edges are corrected at the higher level.

During later stages, large changes to the surface occur, as seen in Figs. 8(e2,f2). However, these areas no longer affect the measured irradiance as they either do not receive any light or the light refracted by this part of the lens fails to reach the target screen.

The final results of MITSUBA (Fig. 8(f6)) and LightTools (Fig. 8(c1)), appear visually very similar. However, we see a structural offset in the LightTools loss shown in Fig. 8(b1). The loss decreases similarly for both ray tracers, though their difference increases in the final batches, similar to the zero-étendue case. Using an extended source emitting a total flux of 1 W, a total of 0.84 W gets through the lens, considering Fresnel losses of which 0.74 W ends up in the target distribution. The 0.16 W, which does not go through the lens, is reflected in the source as discussed in Section 5C of the Supplement 1.

At the highest refinement level, the size of a knot span is 31.25×31.25 μm^2. As we use splines of degree 3, the basis functions are three times the knot span. Therefore, the smallest change in the surface is approximately 95×95 μm^2, which is 160 times larger than the wavelength of the light is 0.55 μm. Therefore, we can assume that diffraction effects will be negligible.

4. Conclusions

We demonstrated a technique to optimize freeform lenses for both zero-étendue sources and finite étendue sources. We accomplished this by utilizing the algorithmic differentiable non-sequential ray tracer MITSUBA 3, giving us access to gradients of the parameters of the freeform surface. This, combined with THB-splines, allows us to gradually increase the degrees of freedom to avoid getting trapped in undesirable local minima. The L_1 norm of the gradient of the vertices within a knot span was used to determine where refinement was necessary.

The method can find a freeform to accurately generate a prescribed target distribution for zero- and finite étendue sources. However, it is not as effective in finding solutions for zero-étendue sources as dedicated zero-etendue solves such as those which solve the Monge-Ampere equation [3]. These are capable of finding similar solutions in a much shorter time. The method excels in its adaptability, as it does not require significant changes to the algorithm to optimize for extended sources.

It is shown that a refinement strategy can significantly improve the design stability compared to simply initiating a freeform with many control points. In addition, using THB-splines allows the designer to keep control over where, on the freeform, new degrees of freedom can be added and obtain similar results to classical B-spline refinement with fewer degrees of freedom.

While optimizing the freeform, the gradient vertexes show interesting information, allowing us to understand how the optimization changes the surface and identify sensitive areas of the freeform.

Future work will focus on further developing the refinement strategy, such as investigating other metrics to determine which knot spans should be refined and a more efficient way to determine the refinement parameters. More freedom in modeling the freeform should be added so that it is not limited to a rectangular optimization domain. Having a more accurate way to transfer the THB-description between programs by using standardized file formats supported by LightTools, such as STEP or IGES.

Funding. Nederlandse Organisatie voor Wetenschappelijk Onderzoek (P15-36).

Acknowledgments. We want to thank Wilbert IJzerman for his input on the manuscript and *Mauritshuis, Den Haag* for allowing us to use the image of the girl with the pearl earring.

Disclosures. The authors declare no conflicts of interest.

Optics EXPRESS

Data availability. Data underlying the results presented in this paper are not publicly available at this time but may be obtained from the authors upon reasonable request.

Supplemental document. See Supplement 1 for supporting content.

References

1. R. Wu, L. Xu, P. Liu, *et al.*, "Freeform illumination design: a nonlinear boundary problem for the elliptic monge–ampére equation," Opt. Lett. **38**(2), 229–231 (2013).
2. N. K. Yadav, J. ten Thije Boonkkamp, and W. IJzerman, "Computation of double freeform optical surfaces using a monge–ampère solver: Application to beam shaping," Opt. Commun. **439**, 251–259 (2019).
3. L. B. Romijn, J. H. ten Thije Boonkkamp, and W. L. IJzerman, "Freeform lens design for a point source and far-field target," J. Opt. Soc. Am. A **36**(11), 1926–1939 (2019).
4. L. Romijn, "Generated jacobian equations in freeform optical design: Mathematical theory and numerics," Ph.D. thesis, Mathematics and Computer Science (2021). Proefschrift.
5. M. J. H. Anthonissen, L. B. Romijn, J. H. M. ten Thije Boonkkamp, *et al.*, "Unified mathematical framework for a class of fundamental freeform optical systems," Opt. Express **29**(20), 31650 (2021).
6. C. Bösel and H. Gross, "Ray mapping approach for the efficient design of continuous freeform surfaces," Opt. Express **24**(13), 14271–14282 (2016).
7. K. Desnijder, P. Hanselaer, and Y. Meuret, "Ray mapping method for off-axis and non-paraxial freeform illumination lens design," Opt. Lett. **44**(4), 771–774 (2019).
8. V. Oliker, "Controlling light with freeform multifocal lens designed with supporting quadric method (sqm)," Opt. Express **25**(4), A58–A72 (2017).
9. Y. Luo, Z. Feng, Y. Han, *et al.*, "Design of compact and smooth free-form optical system with uniform illuminance for led source," Opt. Express **18**(9), 9055–9063 (2010).
10. S. Wei, Z. Zhu, and D. Ma, "Efficient and compact freeform optics design for customized led lighting," Opt. Laser Technol. **167**, 109775 (2023).
11. S. Wei, Z. Zhu, W. Li, *et al.*, "Compact freeform illumination optics design by deblurring the response of extended sources," Opt. Lett. **46**(11), 2770–2773 (2021).
12. S. Hu, K. Du, T. Mei, *et al.*, "Ultra-compact led lens with double freeform surfaces for uniform illumination," Opt. Express **23**(16), 20350–20355 (2015).
13. E. V. Byzov, S. V. Kravchenko, M. A. Moiseev, *et al.*, "Optimization method for designing double-surface refractive optical elements for an extended light source," Opt. Express **28**(17), 24431–24443 (2020).
14. Z. Zhu, S. Wei, Z. Fan, *et al.*, "Freeform illumination optics design for extended led sources through a localized surface control method," Opt. Express **30**(7), 11524–11535 (2022).
15. G. Côté, J.-F. Lalonde, and S. Thibault, "Deep learning-enabled framework for automatic lens design starting point generation," Opt. Express **29**(3), 3841–3854 (2021).
16. Y. Nie, J. Zhang, R. Su, *et al.*, "Freeform optical system design with differentiable three-dimensional ray tracing and unsupervised learning," Opt. Express **31**(5), 7450–7465 (2023).
17. C. Wang, N. Chen, and W. Heidrich, "do: A differentiable engine for deep lens design of computational imaging systems," IEEE Trans. Comput. Imaging **8**, 905–916 (2022).
18. Q. Sun, C. Wang, F. Qiang, *et al.*, "End-to-end complex lens design with differentiable ray tracing," ACM Trans. Graph **40**, 1–13 (2021).
19. B. d. Koning, A. Heemels, A. Adam, *et al.*, "Gradient descent-based freeform optics design for illumination using algorithmic differentiable non-sequential ray tracing," Optimization and Engineering pp. 1–33 (2023).
20. H. Wang, Y. Luo, H. Li, *et al.*, "Extended ray-mapping method based on differentiable ray-tracing for non-paraxial and off-axis freeform illumination lens design," Opt. Express **31**(19), 30066–30078 (2023).
21. L. Li and X. Hao, "Optimizing triangle mesh lenses for non-uniform illumination with an extended source," Opt. Lett. **48**(7), 1726–1729 (2023).
22. W. Jakob, S. Speierer, N. Roussel, *et al.*, "Dr. jit: a just-in-time compiler for differentiable rendering," ACM Trans. Graph. **41**(4), 1–19 (2022).
23. L. Piegl and W. Tiller, *The NURBS book* (Springer Science & Business Media, 1996).
24. T. Dokken, T. Lyche, and K. F. Pettersen, "Polynomial splines over locally refined box-partitions," Comput. Aided Geom. Des. **30**(3), 331–356 (2013).
25. D. R. Forsey and R. H. Bartels, "Hierarchical b-spline refinement," in *Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, (1988), pp. 205–212.
26. C. Giannelli, B. Jüttler, and H. Speleers, "Thb-splines: The truncated basis for hierarchical splines," Comput. Aided Geom. Des. **29**(7), 485–498 (2012).
27. D. C. Thomas, L. Engvall, S. K. Schmidt, *et al.*, "U-splines: Splines over unstructured meshes," Computer Methods in Applied Mechanics and Engineering **401**, 115515 (2022).
28. T. W. Sederberg, J. Zheng, A. Bakenov, *et al.*, "T-splines and t-nurccs," ACM Trans. Graph. **22**(3), 477–484 (2003).
29. E. Bailey and S. Carayon, "Beyond nurbs: enhancement of local refinement through t-splines," in *Nonimaging Optics and Efficient Illumination Systems IV*, vol. 6670 (SPIE, 2007), pp. 151–160.
30. S. I. Chandra and A. Shalom, *Intelligent Freeform Deformation for LED Illumination Optics*, vol. 16 (KIT Scientific Publishing, 2018).

Optics EXPRESS

31. C. Giannelli, B. Jüttler, S. K. Kleiss, *et al.*, "Thb-splines: An effective mathematical technology for adaptive refinement in geometric design and isogeometric analysis," Computer Methods in Applied Mechanics and Engineering **299**, 337–365 (2016).
32. J. Nocedal and S. J. Wright, Numerical optimization (2006).
33. A. Griewank and A. Walther, *Evaluating derivatives: principles and techniques of algorithmic differentiation* (SIAM, 2008).
34. S. Zhao, W. Jakob, and T.-M. Li, "Physics-based differentiable rendering: from theory to implementation," in *ACM siggraph 2020 courses*, (2020), pp. 1–30.
35. A. Paszke, S. Gross, F. Massa, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*, (Curran Associates, Inc., 2019), pp. 8024–8035.
36. D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv, arXiv:1412.6980 (2014).
37. Synopsys, Lighttools, (2023). Https://www.synopsys.com/optical-solutions/lighttools.html.
38. M. N. Ricketts, R. Winston, and V. Oliker, "Diffraction effects in freeform optics," in *Nonimaging Optics: Efficient Design for Illumination and Solar Concentration XII*, vol. 9572 (SPIE, 2015), pp. 126–131.
39. S. Zwick, R. Feßler, J. Jegorov, *et al.*, "Resolution limitations for tailored picture-generating freeform surfaces," Opt. Express **20**(4), 3642–3653 (2012).
40. M. Brand, "Minimum spot size and maximum detail in extended-source freeform illumination," in *OSA Optical Design and Fabrication 2021 (Flat Optics, Freeform, IODC, OFT)*, (Optica Publishing Group, Washington, DC, 2021), p. JTh1A.1.
41. A. N. Heemels, A. J. Adam, and H. P. Urbach, "Limits of realizing irradiance distributions with shift-invariant illumination systems and finite étendue sources," J. Opt. Soc. Am. A **40**(7), 1289–1302 (2023).