MEP Literature review

Designing free-form optics for multiple source illumination using differentiable ray-tracing and neural networks

Bart de Koning, BSc

Delft University of Technology, The Netherlands

March 11, 2021

TUDelft

Contents

Caustic design

Caustic design

Def: *A part of non-imaging optics concerned with the geometrical design of optical components to transform the incident illumination from light sources into a desired target distribution.*[1]

Figure: An example of caustic design: approximating the painting *A Sunday Afternoon on the Island of La Grande Jatte* by Georges Seurat.[2]

---

[1] J. Meyron, Q. Mérigot, and B. Thibert (2018). "Light in Power: A general and parameter-free algorithm for caustic design". In: *ACM Transactions on Graphics* 37 (6). ISSN: 15577368. DOI: 10.1145/3272127.3275056

[2] M. Nimier-David, D. Vicini, T. Zeltner, and W. Jakob (Dec. 2019b). "Mitsuba 2: A Retargetable Forward and Inverse Renderer". In: *Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 38.6. DOI: 10.1145/3355089.3356498

Previous work

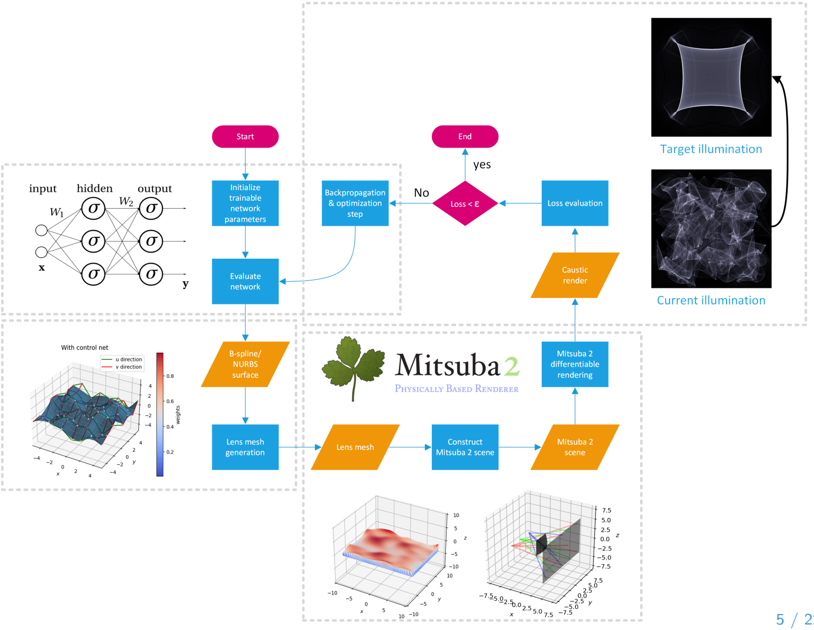
- Lucas' work on Caustic design with MLP + B-spline surface + wave optics.[3]
- Differentiable rendering in *Mitsuba 2*.[4]

[3] L. H. Crijns (2021). *PINN inspired Freeform Design*. Delft University of Technology

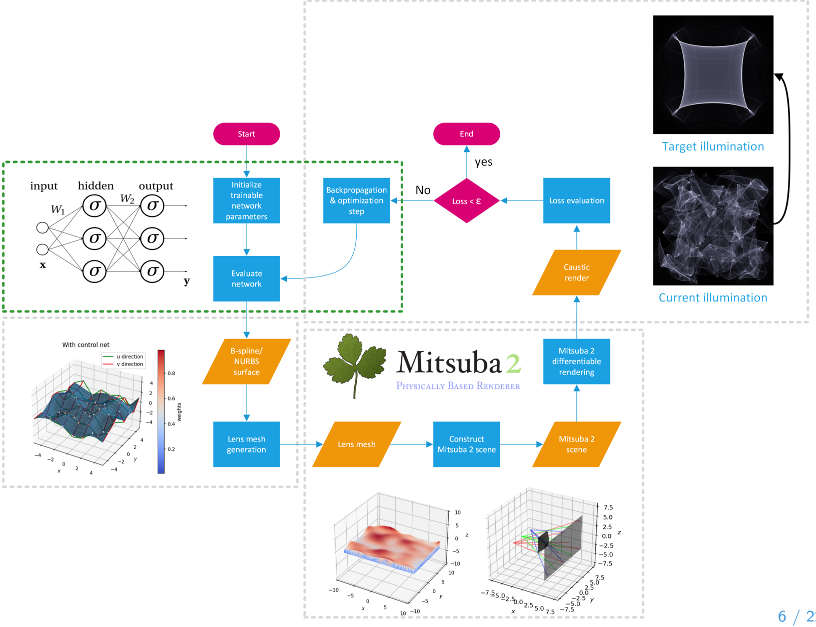
[4] M. Nimier-David, D. Vicini, T. Zeltner, and W. Jakob (2019a). "Mitsuba 2". In: *ACM Transactions on Graphics* 38 (6), pp. 1–17. ISSN: 0730-0301. DOI: 10.1145/3355089.3356498

TUDelft

Optimization pipeline



Scientific Machine Learning



Scientific Machine Learning: PINNs

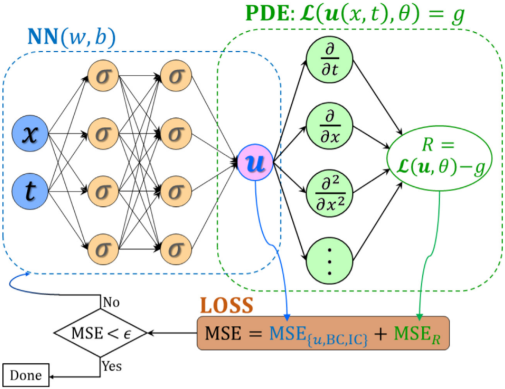
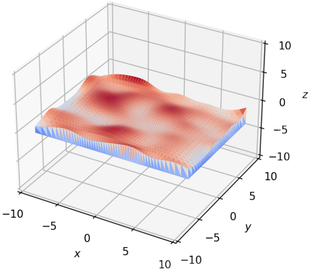


Figure: PINN schematic.[5]

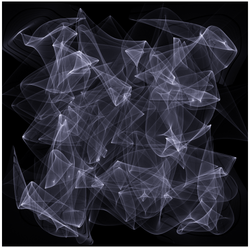
[5] X. Meng, Z. Li, D. Zhang, and G. E. Karniadakis (Sept. 2019). "PPINN: Parareal Physics-Informed Neural Network for time-dependent PDEs". In: DOI: 10.1016/j.cma.2020.113250

(Scientific Machine Learning): Inverse problem

- Forward problem: Lens ⇒ caustic
- Inverse problem: Lens ⇐ caustic

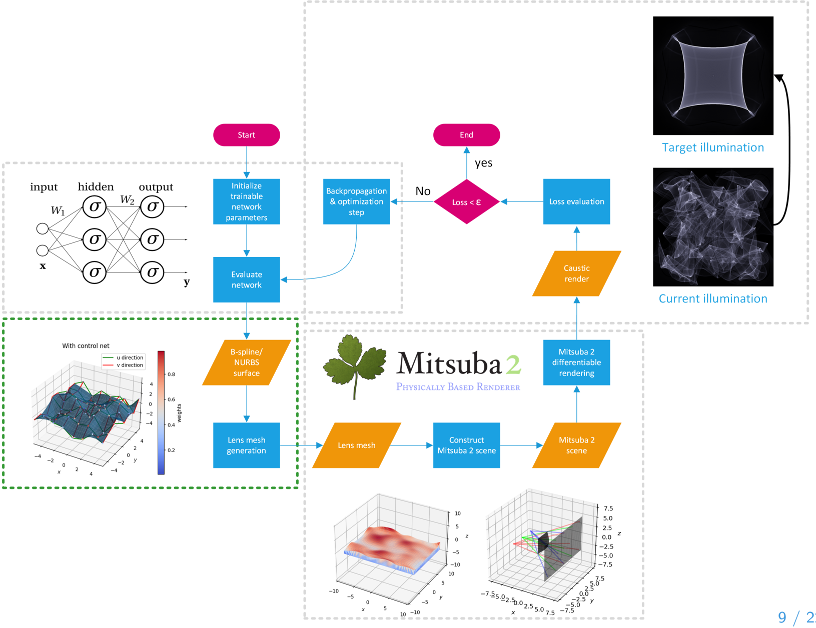


⇔

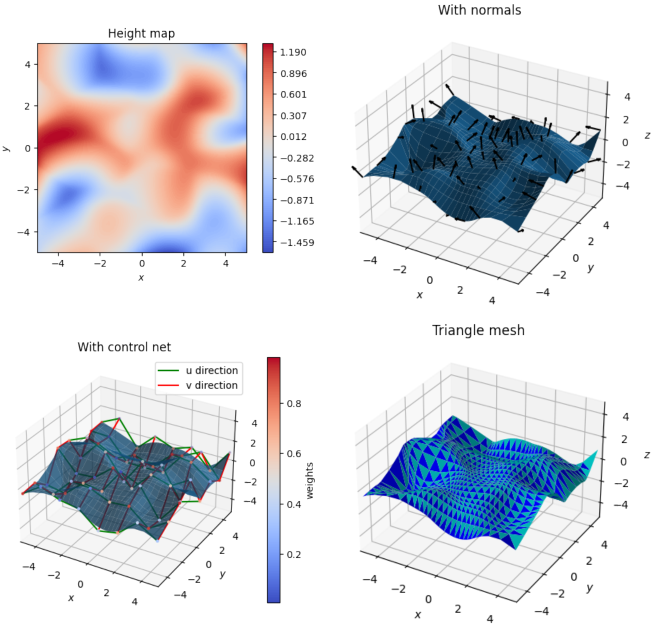


⇒ Analysis-by-synthesis

Spline surfaces



Spline surfaces: Plots



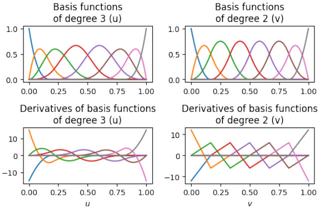
Spline surfaces: Definitions

B-spline surface:

$$\mathbf{S}(u,v) := \sum_{i=0}^{n_1}\sum_{j=0}^{n_2} N_{i,p}(u)N_{j,q}(v)\mathbf{P}_{i,j} \tag{1}$$

NURBS surface:

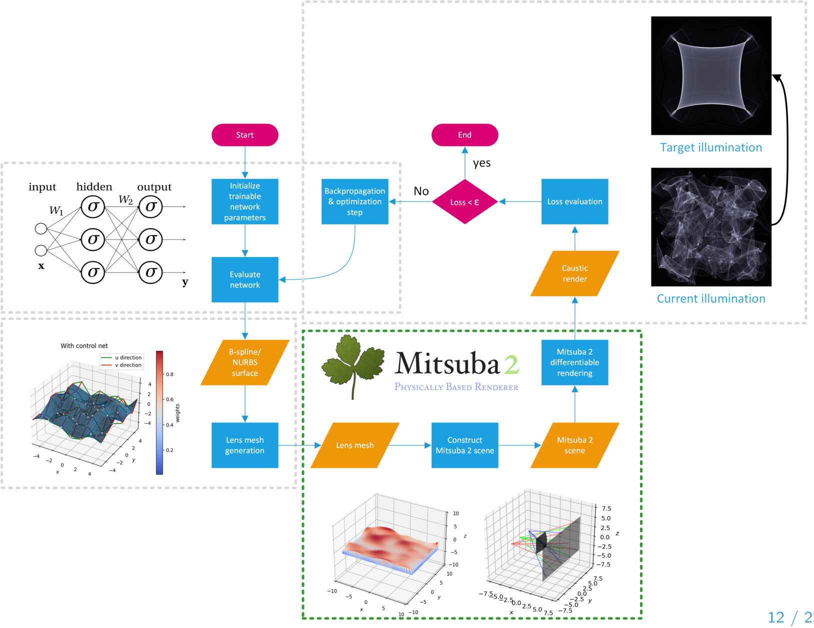
$$\mathbf{S}(u,v) := \sum_{i=0}^{n_1}\sum_{j=0}^{n_2} \frac{w_{i,j}N_{i,p}(u)N_{j,q}(v)}{\sum_{i'=0}^{n}\sum_{j'=0}^{m} w_{i',j'}N_{i',p}(u)N_{j',q}(v)}\mathbf{P}_{i,j} \tag{2}$$



Standard work on NURBS: *NURBS book*.[6]

[6] L. Piegl and W. Tiller (1997). *The NURBS Book*. Springer Berlin Heidelberg. ISBN: 978-3-540-61545-3. DOI: 10.1007/978-3-642-59223-2

Ray-tracing



Start
Initialize trainable network parameters
Evaluate network
input
hidden
output
W_1
W_2
σ
x
y
B-spline/ NURBS surface
Lens mesh generation
With control net
u direction
v direction
weights
Lens mesh
Construct Mitsuba 2 scene
Mitsuba 2 scene
Mitsuba 2 differentiable rendering
Mitsuba 2
Physically Based Renderer
Caustic render
Loss evaluation
Loss < ε
No
yes
End
Backpropagation & optimization step
Target illumination
Current illumination

Ray-tracing: Overview

Important scene components

- Light source(s)
- Lens mesh
- BSDF: Bi-directional scattering distribution function
- Receiver screen (diffuser)
- Camera

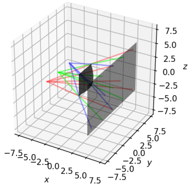


Figure: Schematic setup of a ray-tracer I wrote

Ray-tracing: BSDF

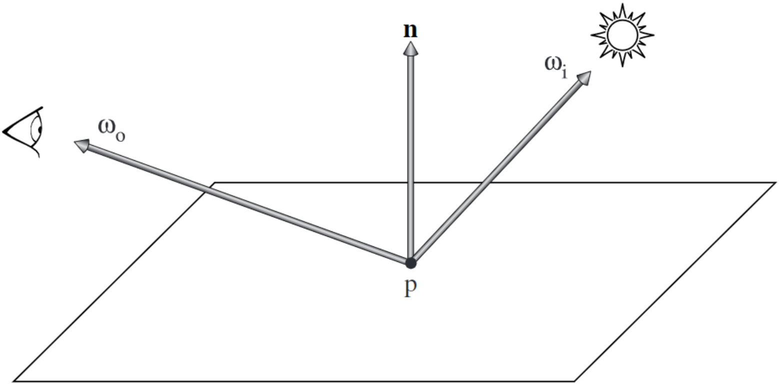
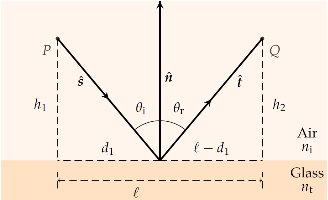


Figure: BSDF: $f(\omega_o, \omega_i)$.[7]

[7] M. Pharr, W. Jakob, and G. Humphreys (2017). *Physically Based Rendering*. DOI: 10.1016/b978-0-12-800645-0.50001-4

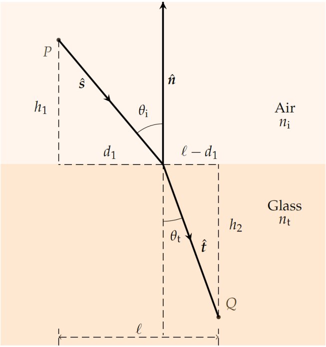
Ray-tracing: Optics basics[8]



(a) **Reflection**

angles: $\theta_i = \theta_r$

vectors: $\hat{\mathbf{t}} = \hat{\mathbf{s}} - 2\langle \hat{\mathbf{s}}, \hat{\mathbf{n}} \rangle \hat{\mathbf{n}}$



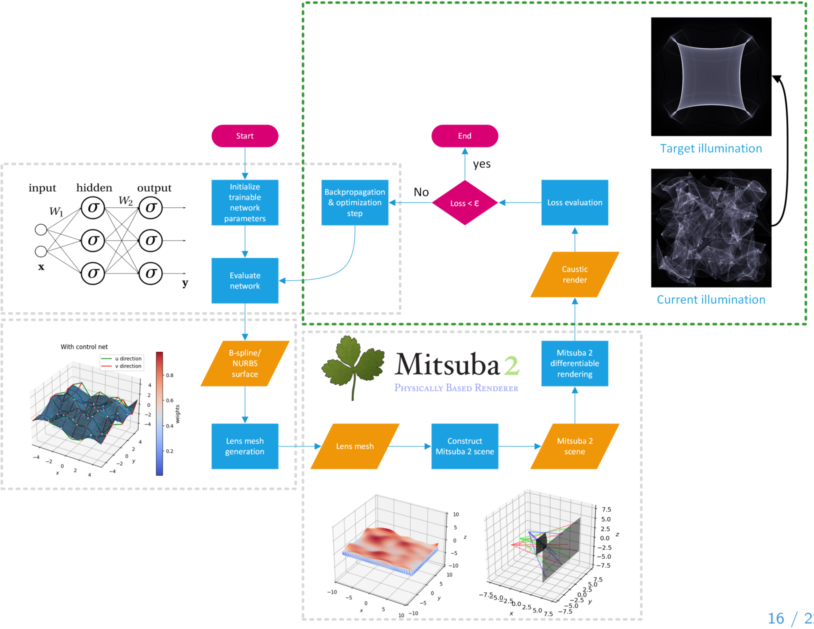
(b) **Refraction**

angles: $n_i \sin\theta_i = n_t \sin\theta_t$

vectors: $\hat{\mathbf{t}} = \frac{n_i}{n_t}\hat{\mathbf{s}} - \left(\frac{n_i}{n_t}\langle \hat{\mathbf{s}}, \hat{\mathbf{n}} \rangle + \sqrt{1 - \left(\frac{n_i}{n_t}\right)^2 (1 - \langle \hat{\mathbf{s}}, \hat{\mathbf{n}} \rangle^2)} \right) \hat{\mathbf{n}}$

[8] L. B. Romijn (2021). *Generated Jacobian Equations in Freeform Optical Design*. ISBN: 9789464166941

Optimization



Optimization: Landscape

Loss = $\|\text{Current illumination} - \text{Target illumination}\|^2$

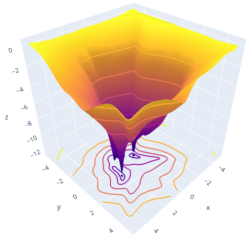


Figure: Example of a design landscape: loss (z) as a function of network parameters (x, y).[9]

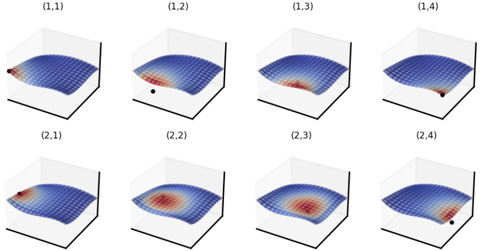
Optimizers: Adam[10], L-BFGS[11]

[9] J. Berner, P. Grohs, G. Kutyniok, and P. Petersen (May 2021). "The Modern Mathematics of Deep Learning". In

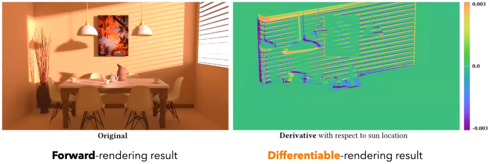
[10] D. P. Kingma and J. Ba (Dec. 2014). "Adam: A Method for Stochastic Optimization". In

[11] D. C. Liu and J. Nocedal (1989). *On the limited memory BFGS method for large scale optimization*. Pp. 503–528

Optimization: Automatic Differentiation



(a) Gradient of the surface with respect to the z-coordinate of a control point.



(b) Differentiable rendering example: derivative w.r.t. sun location. Source: `https://diff-render.org/`.

Figure: AD examples

Optimization: Inverse rendering

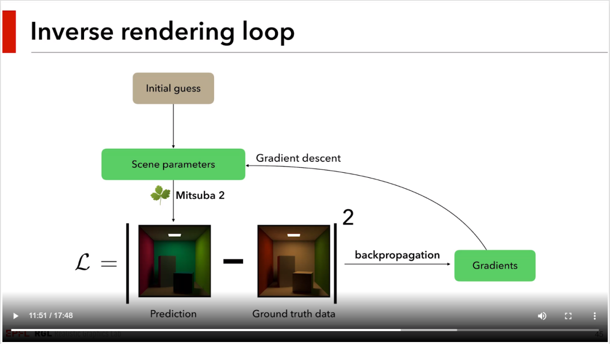


Figure: Inverse rendering without neural network.[12]

[12] M. Nimier-David, D. Vicini, T. Zeltner, and W. Jakob (Dec. 2019b). "Mitsuba 2: A Retargetable Forward and Inverse Renderer". In: *Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 38.6. DOI: 10.1145/3355089.3356498

Research: Considerations

Hyper-parameters

- Network: width, depth, connectivity
- Sources: type and amount
- Spline surface(s): control net size, degrees

(Un)free parameters

- Control point z coordinates
- Control point x, y coordinates
- Control point weights
- Source location(s)

Research: Starting model

Starting model

- One simple (zero-etendue) fixed light source: point or plane
- Specularly scattering lens mesh with one B-spline surface, $n = 1.5$
- A simple MLP with a convolutional layer reflecting the B-spline basis function support overlap, only control point z-coordinates are output
- A loss function consisting of a RMS-difference of the current and target render
- The L-BFGS optimizer

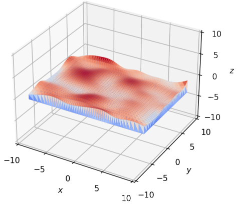


Figure: Example lens mesh.

Research: Goals

Research goals

1. See whether the lens that produces a certain caustic can be found back
2. See what the influence of training with weights is on convergence
3. See whether a lens for desired smooth caustics, *e.g.* for street lighting, can be found (and for which surface hyper-parameters)