SIC Saarland Informatics Campus

SAARBRÜCKEN

EUROGRAPHICS 2023

Learning to Learn and Sample BRDFs

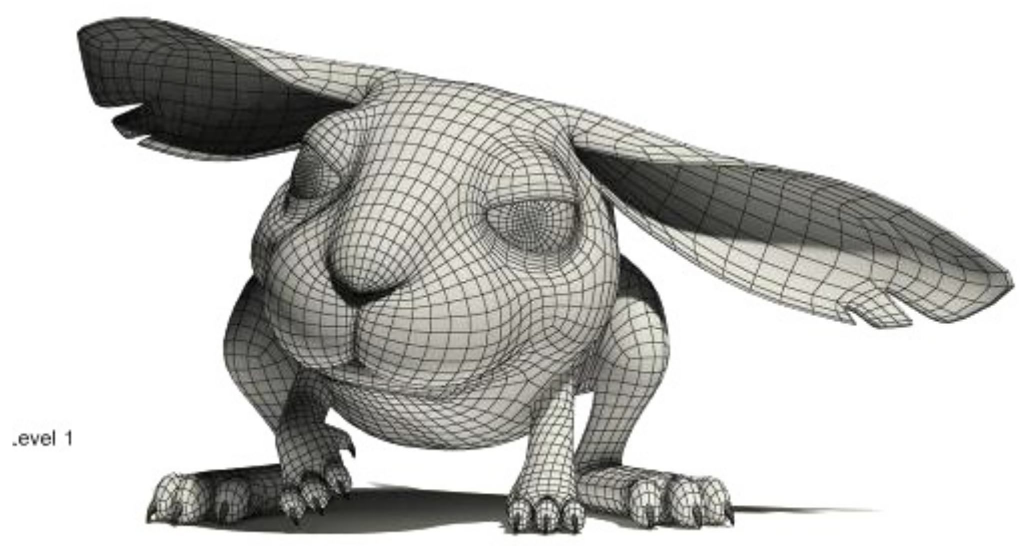
Chen Liu, Michael Fischer, Tobias Ritschel
University College London

Motivation

Assets

Content

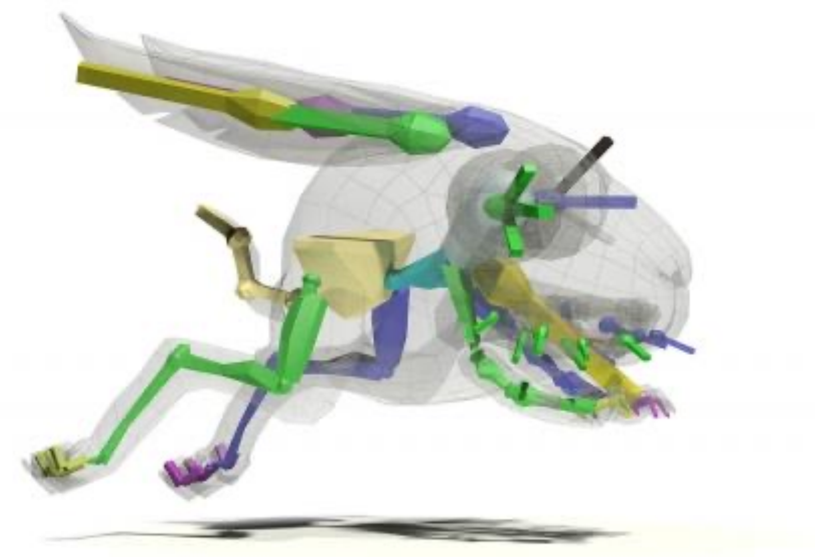
=



+

+

+



Shape

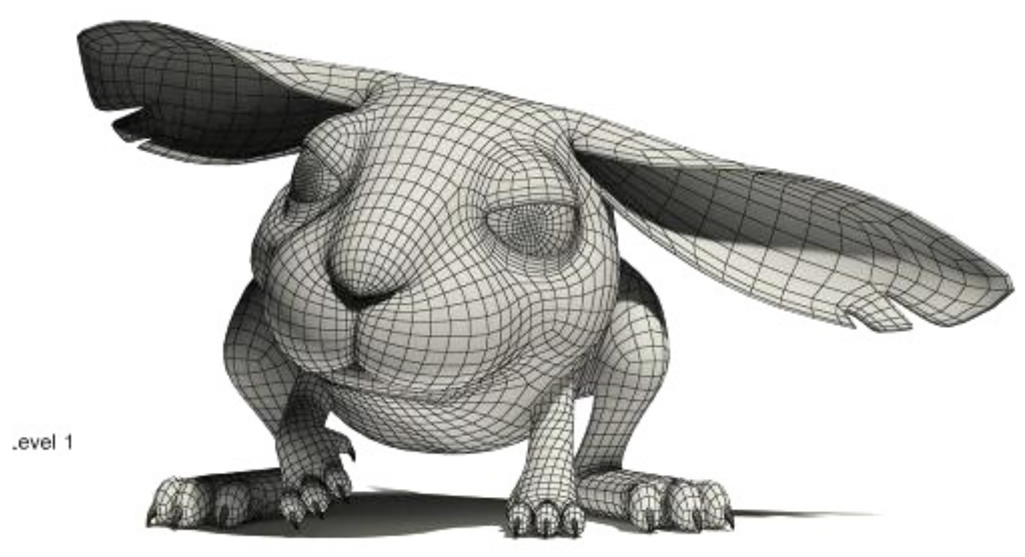
Material

Illumination

Animation

Neural assets

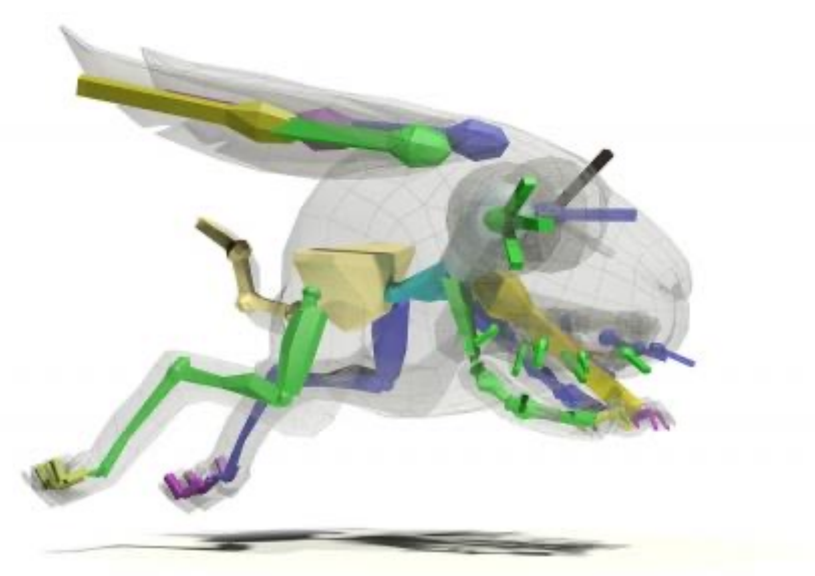
=

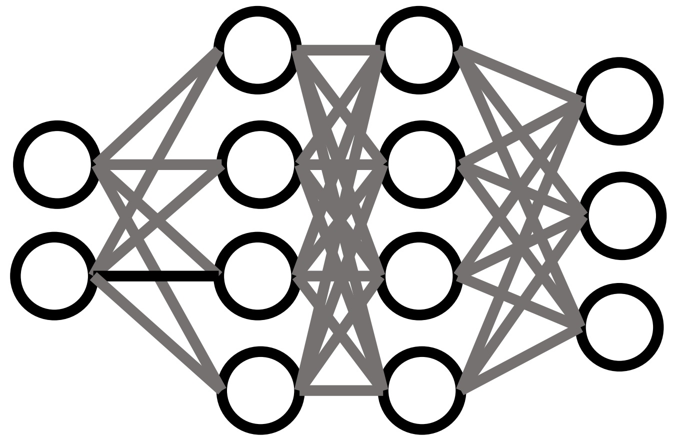


+

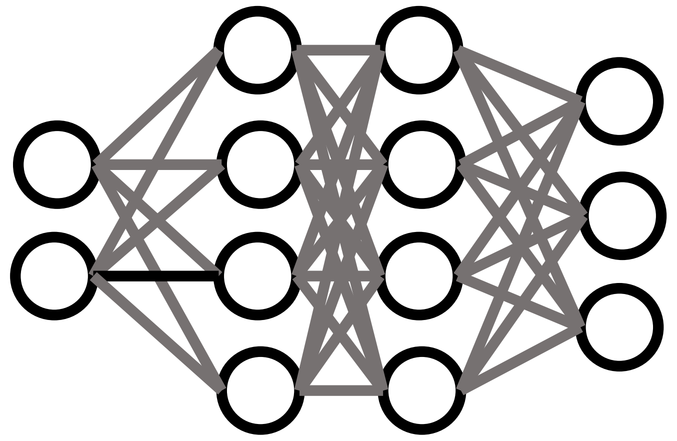
+

+

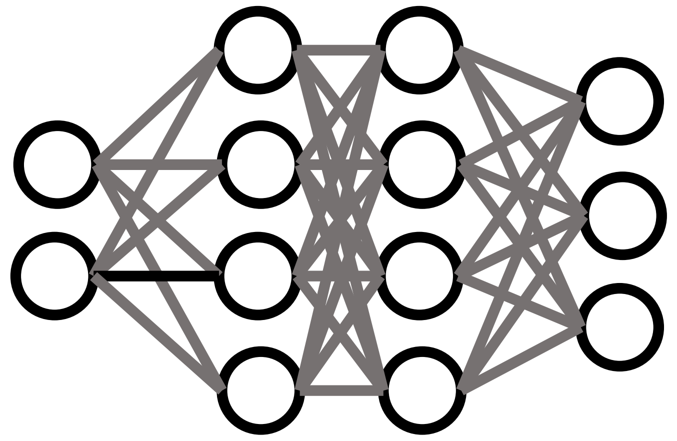




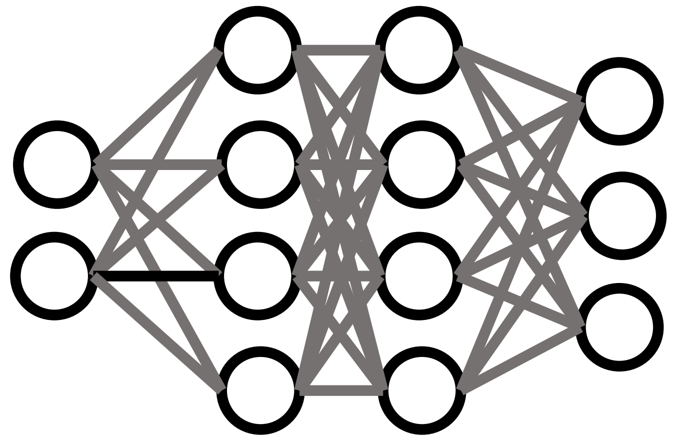
Shape



Material

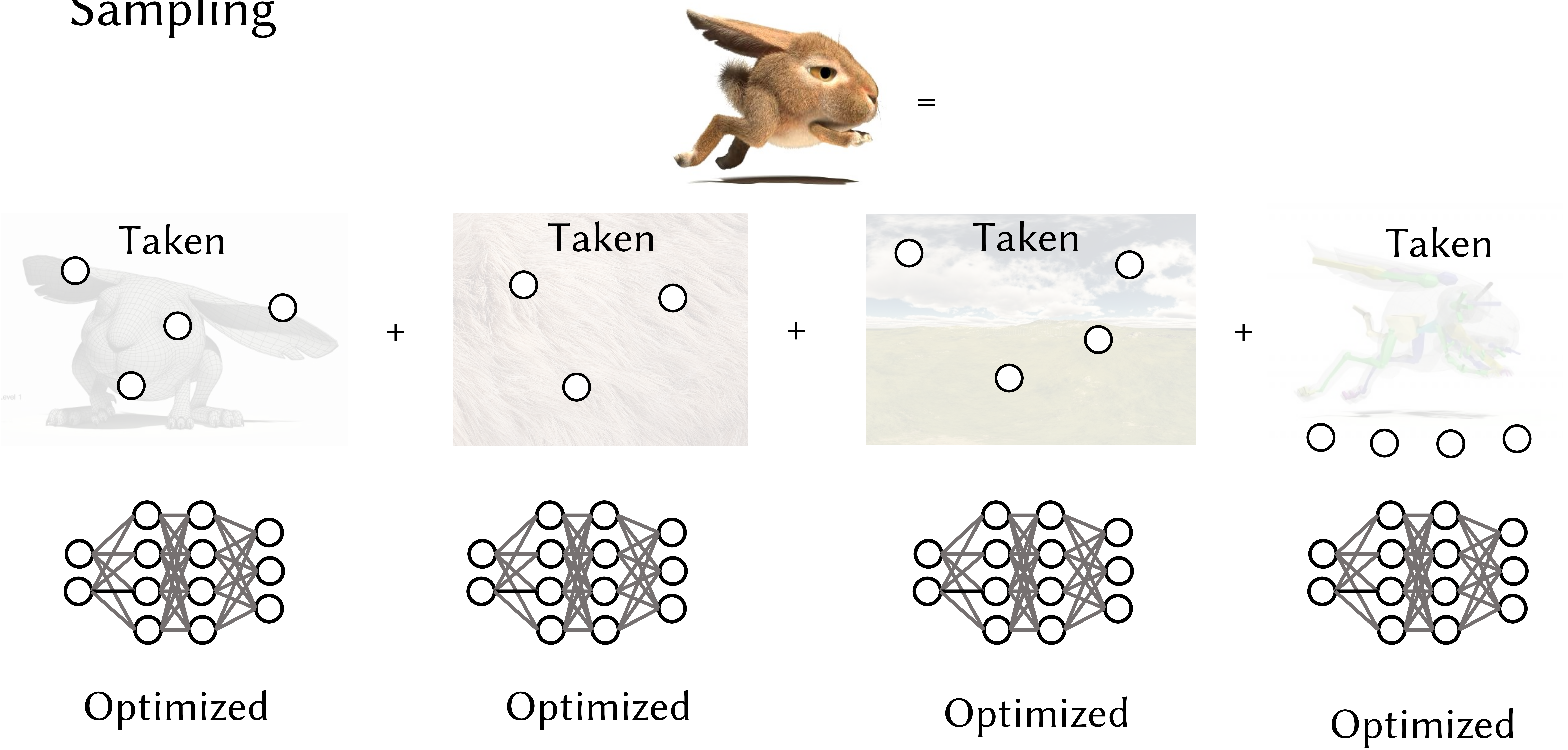


Illumination



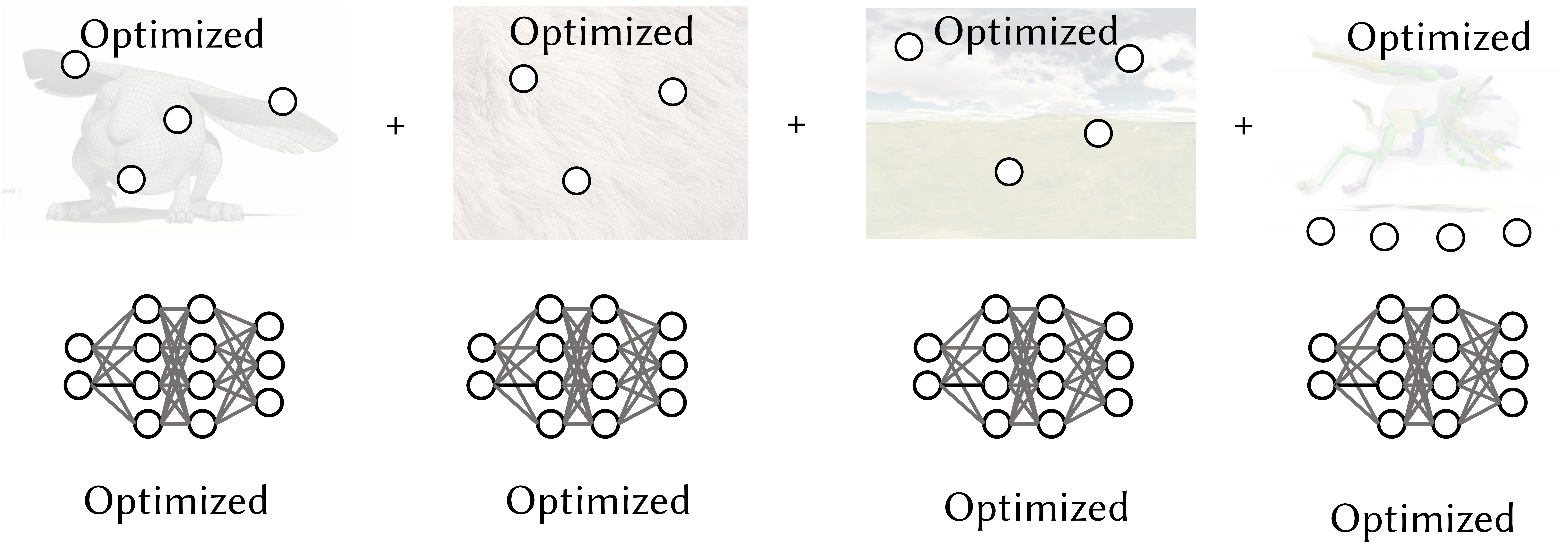
Animation

Sampling

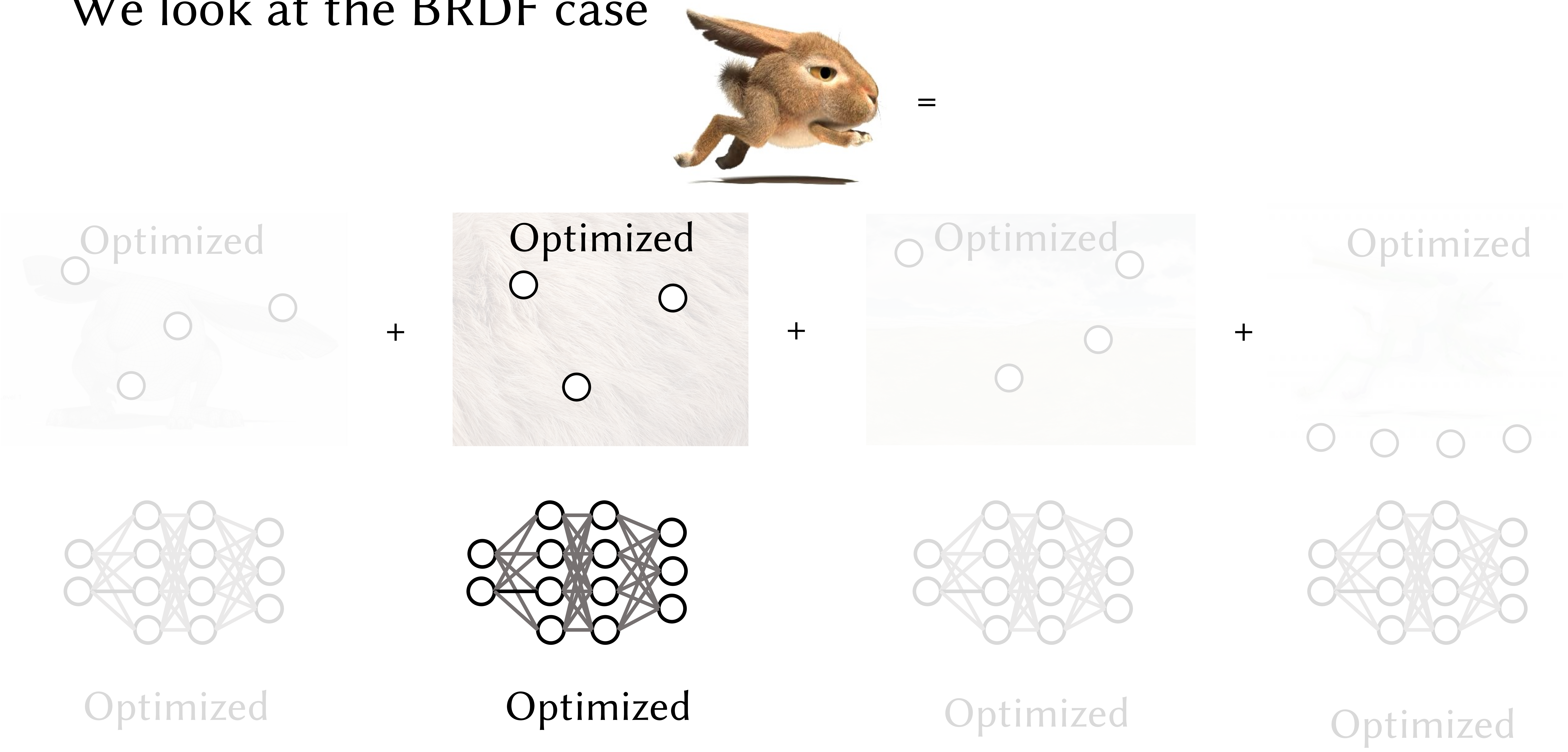

=
Taken
+
Taken
+
Taken
+
Taken
Optimized
Optimized
Optimized
Optimized

Also optimize sampling

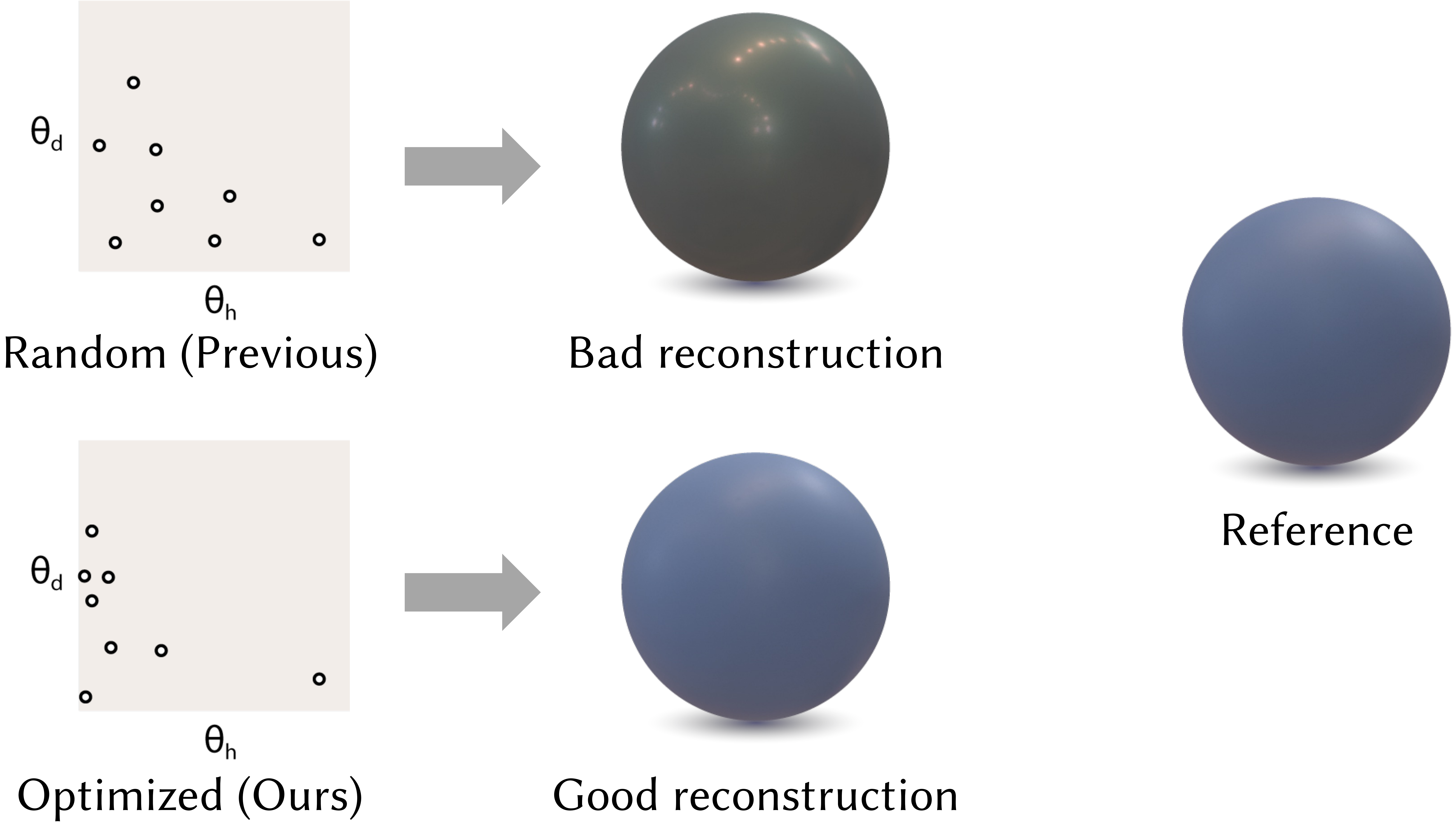
=



We look at the BRDF case



Not all samples are equally useful



Related work

Backbone BRDF models

1) Analytical models

- Phong [PB75]
- Cook-Torrance [CT82]

$$f_r(\omega_i, \omega_o) = \boxed{k_d}\frac{1}{\pi} + \boxed{k_s}\frac{q+2}{2\pi}\langle \omega_o, r \rangle^{\boxed{q}}$$

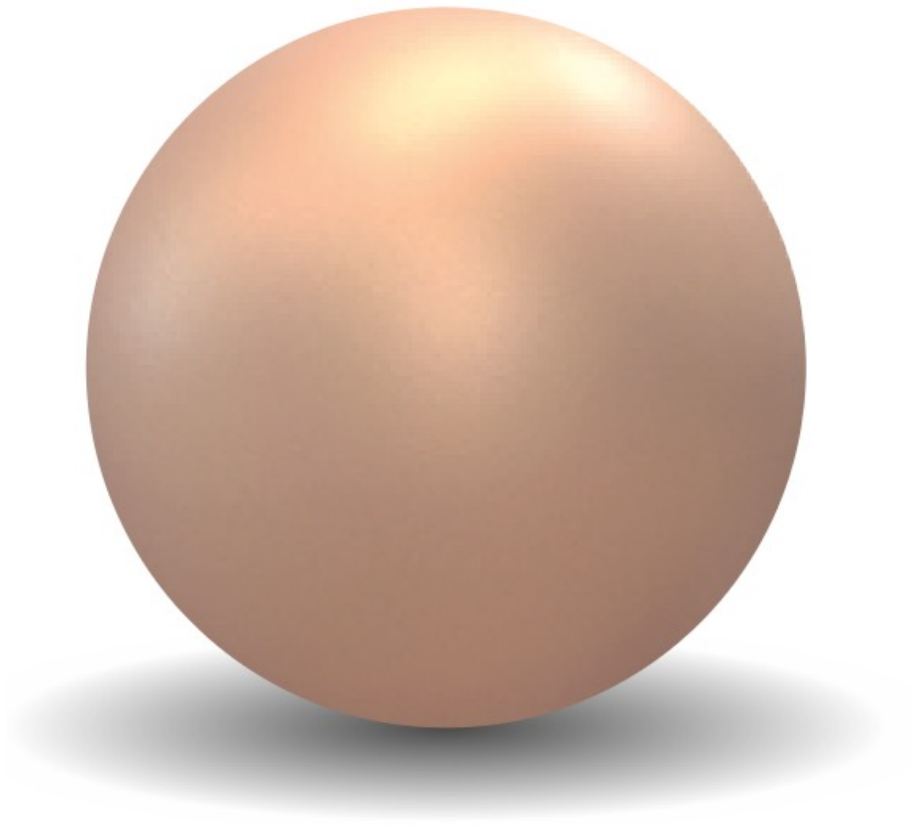
$$f_r(\omega_i, \omega_o) = \boxed{k_d}\frac{1}{\pi} + \boxed{k_s}\frac{D\boxed{(\alpha)}GF(\boxed{F_0})}{\pi \cos(\theta_i)\cos(\theta_o)}$$

Optimize parameters

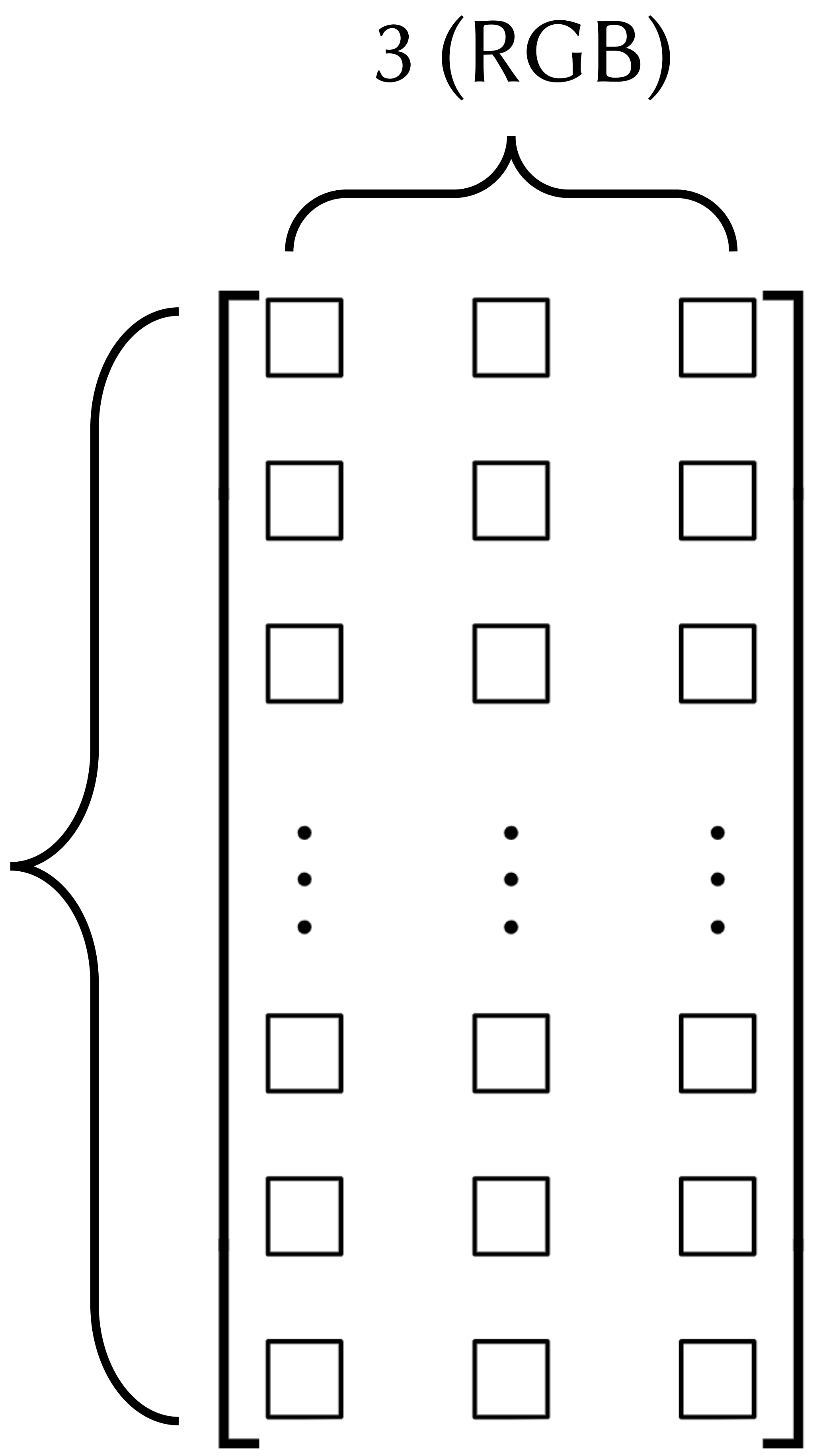
Backbone BRDF models

2) Linear PCA model [Mat03] [NJR15]

- MERL Dataset

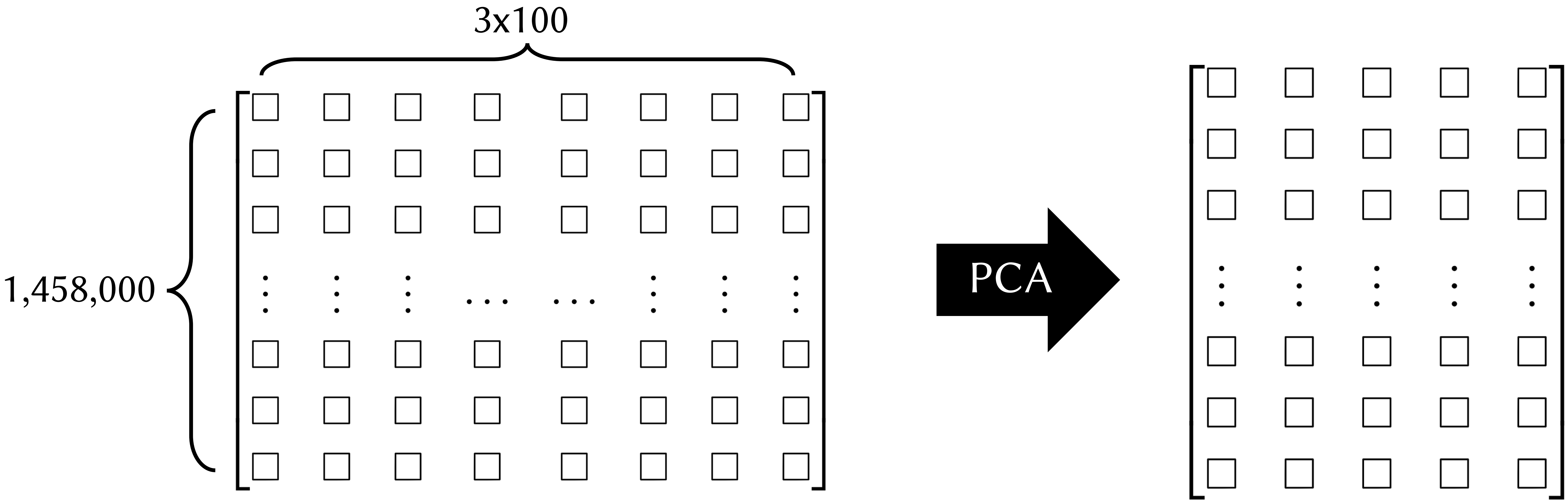


Measured in the resolution of
90 (θ_h) x 90 (θ_d) x 180 (ϕ_d) = 1,458,000



Backbone BRDF models

2) Linear PCA model [Mat03] [NJR15]

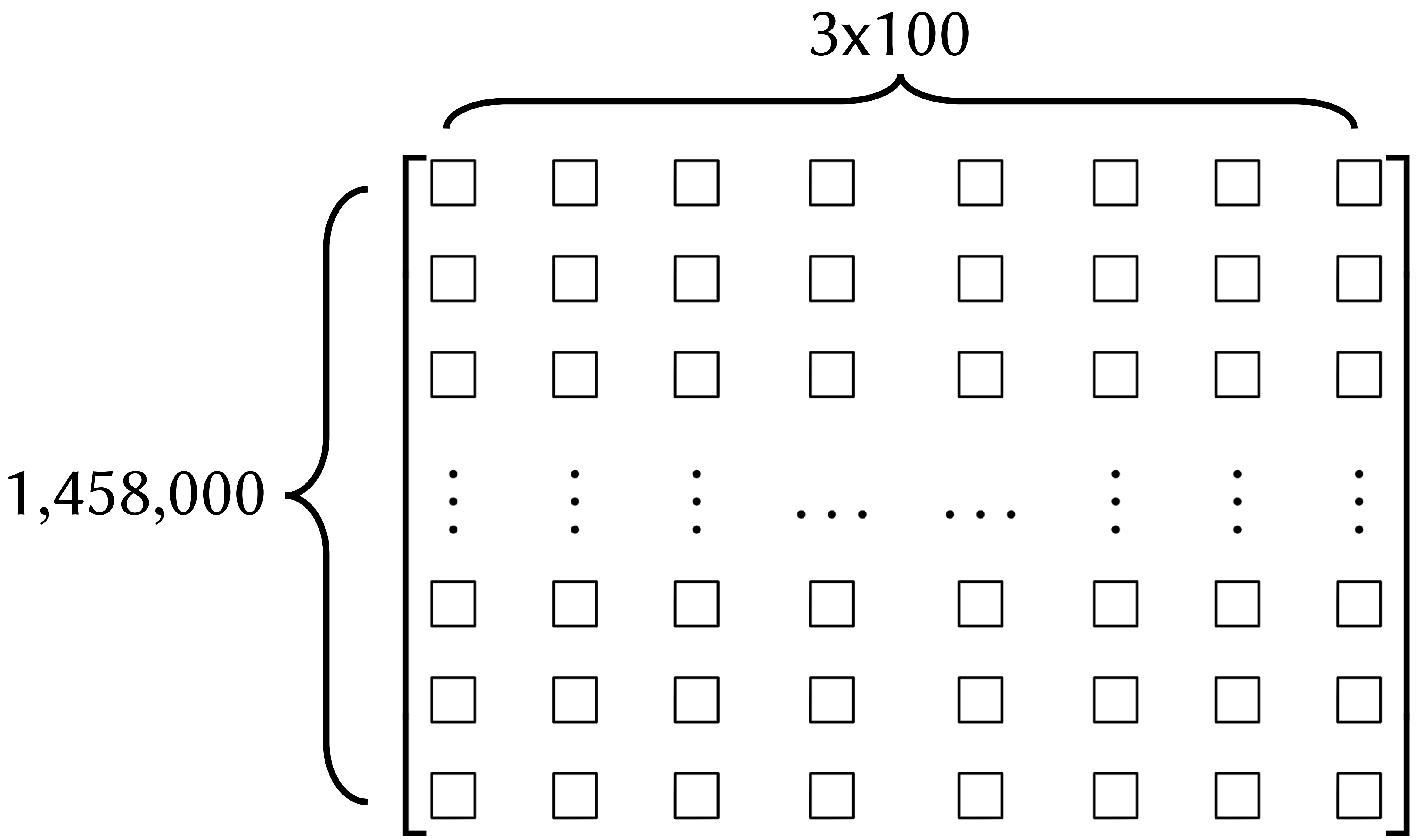


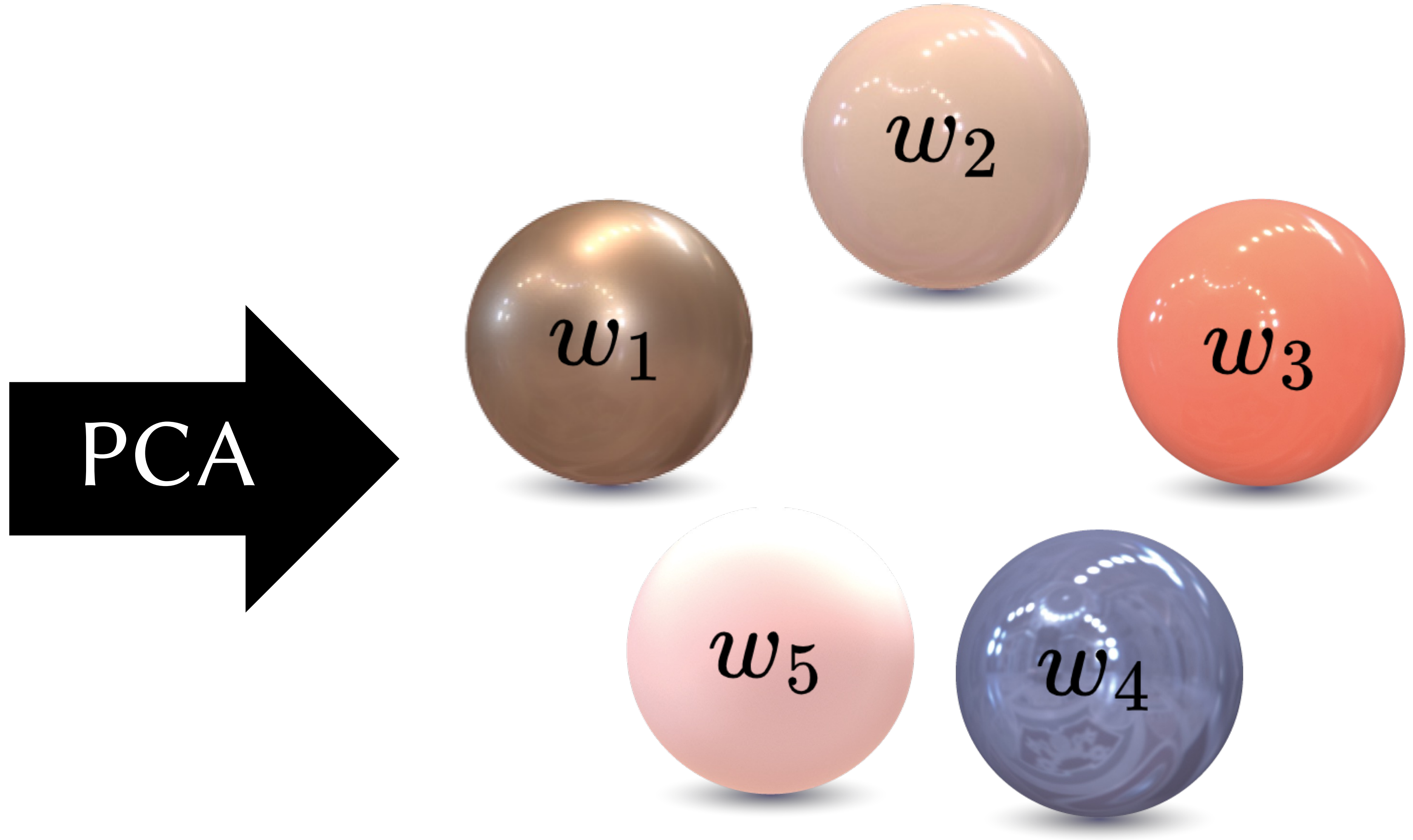
All 100 materials concatenated horizontally

5 Principle Components

Backbone BRDF models

2) Linear PCA model [Mat03] [NJR15]



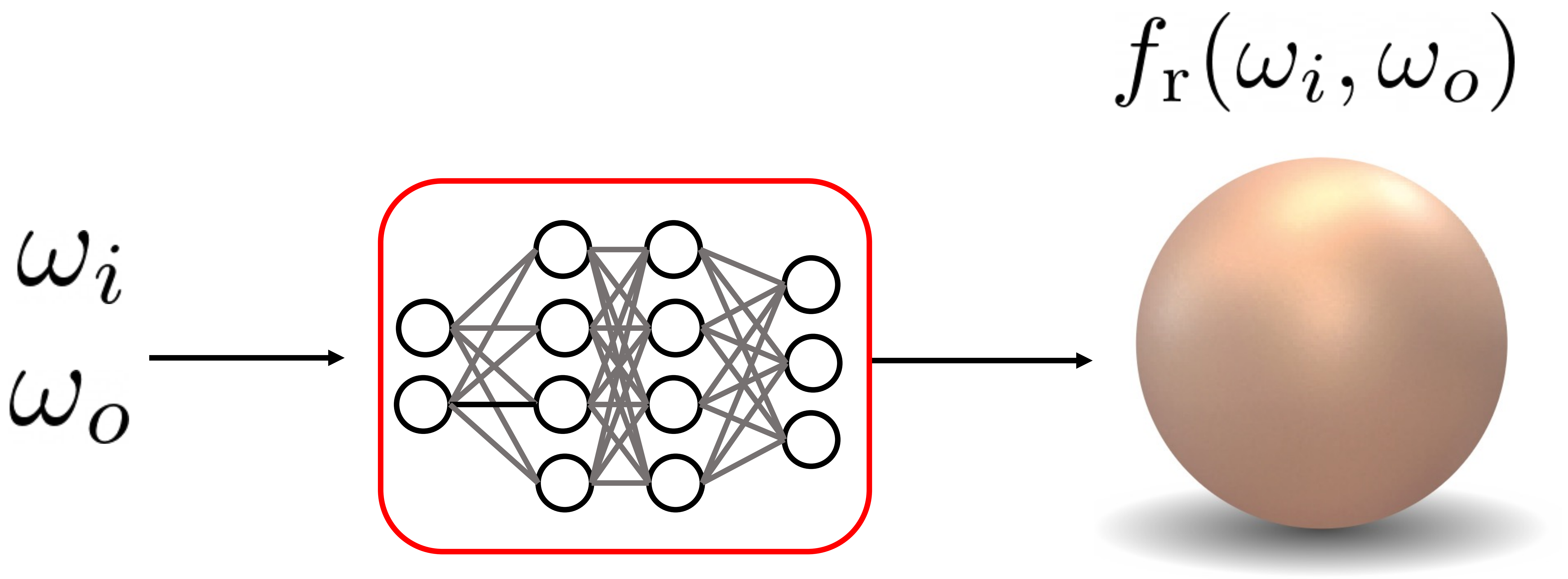


All 100 materials concatenated horizontally

fit weights by Least Squares

Backbone BRDF models

3) Neural BRDF [SRRW21]



Also optimized

Backbone BRDF models

1) Analytical [PB75] [CT82]

2) Linear PCA [Mat03] [NJR15]

3) Neural [SRRW21]

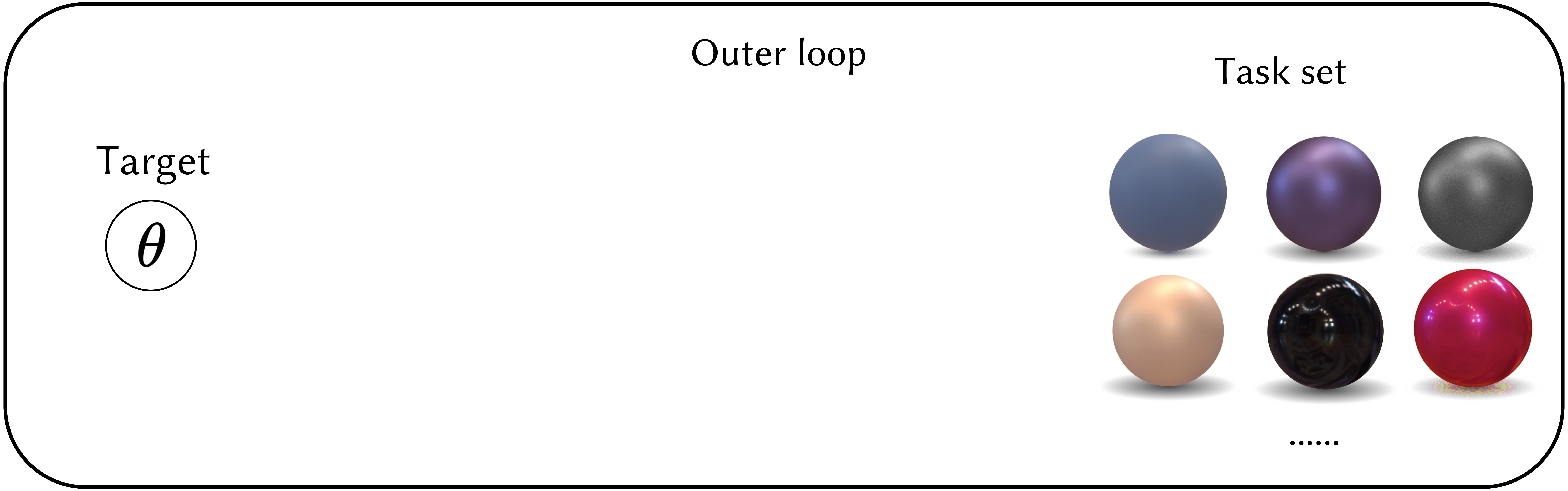
To optimize samples, our method **is orthogonal to ALL** these models

Our solution

Our solution: meta-sampling

Meta learning: learning to learn

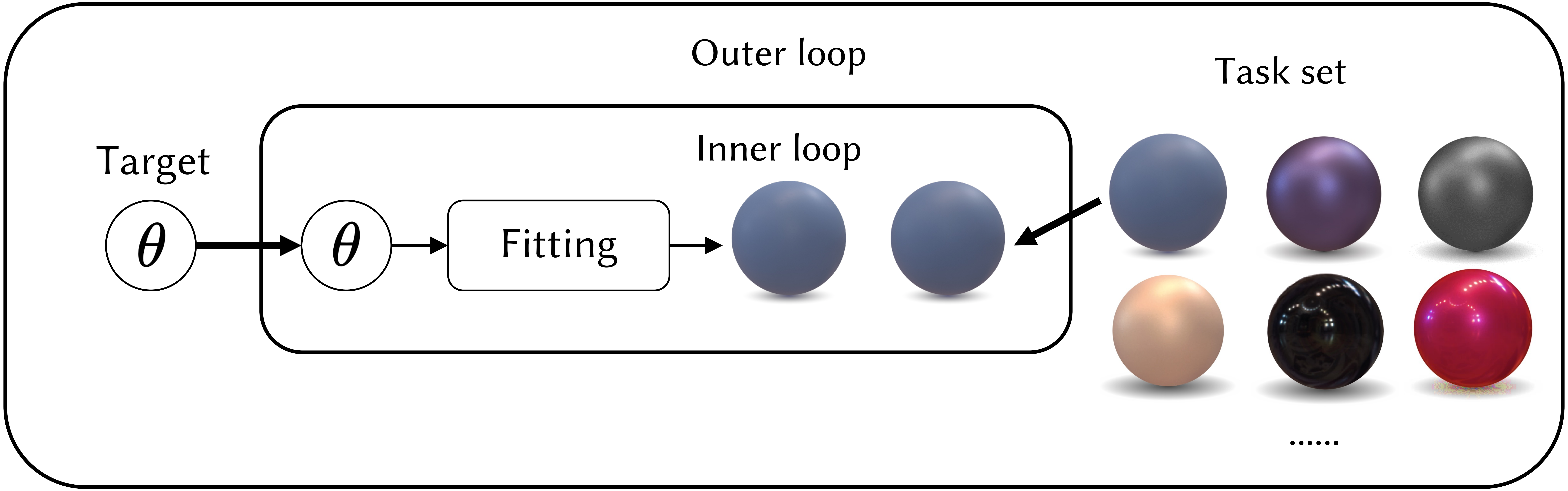
- Learn the commonalities (outer loop) from doing the same type of tasks (inner loop)



Our solution: meta-sampling

Meta learning: learning to learn

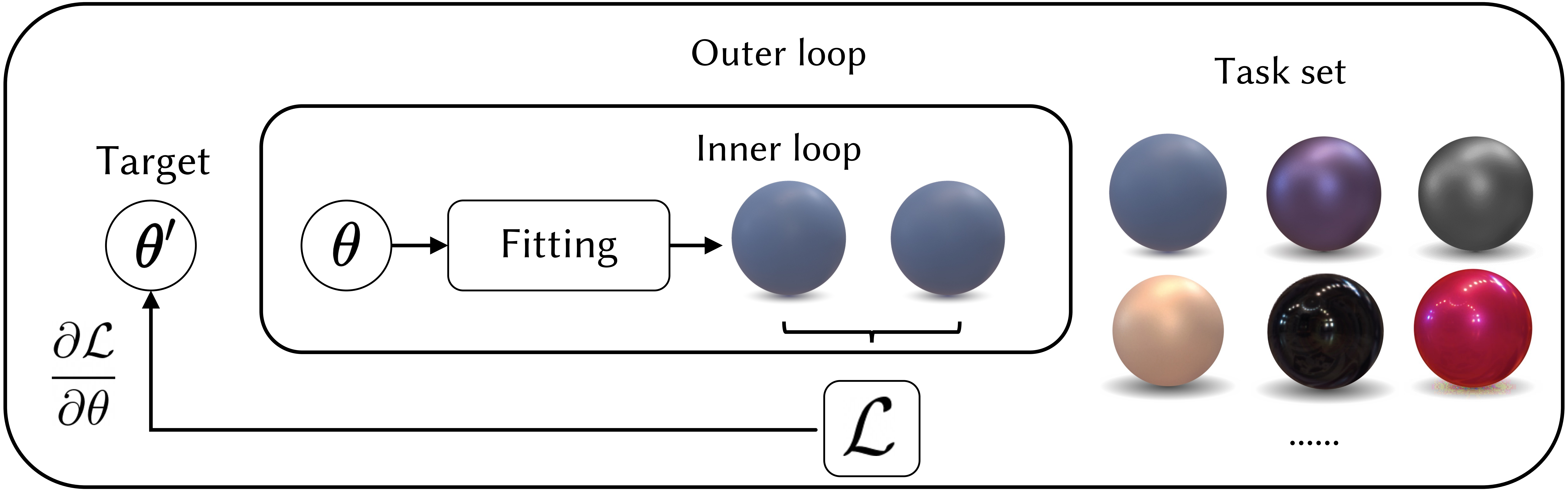
- Learn the commonalities (outer loop) from doing the same type of tasks (inner loop)



Our solution: meta-sampling

Meta learning: learning to learn

- Learn the commonalities (outer loop) from doing the same type of tasks (inner loop)



Our solution: meta-sampling

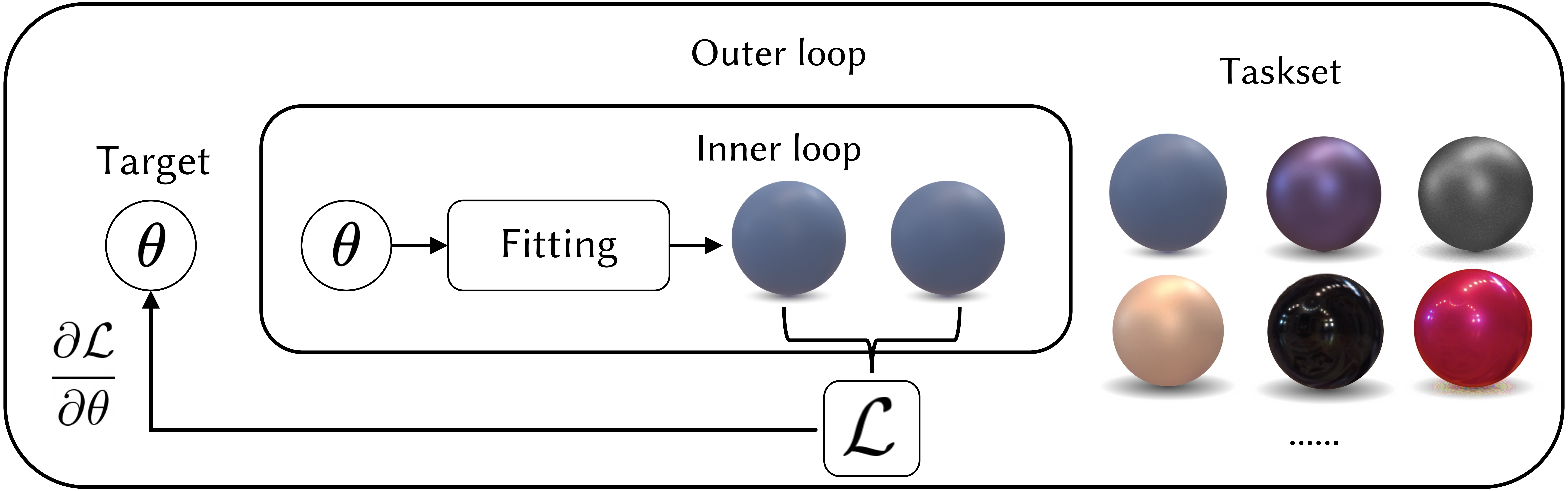
Meta learning in Computer Graphics

- Metappearance [FR22]
- MetaSDF [SCT*20]

Our solution: meta-sampling

Meta learning in Computer Graphics

- Metappearance [FR22]
- MetaSDF [SCT*20]

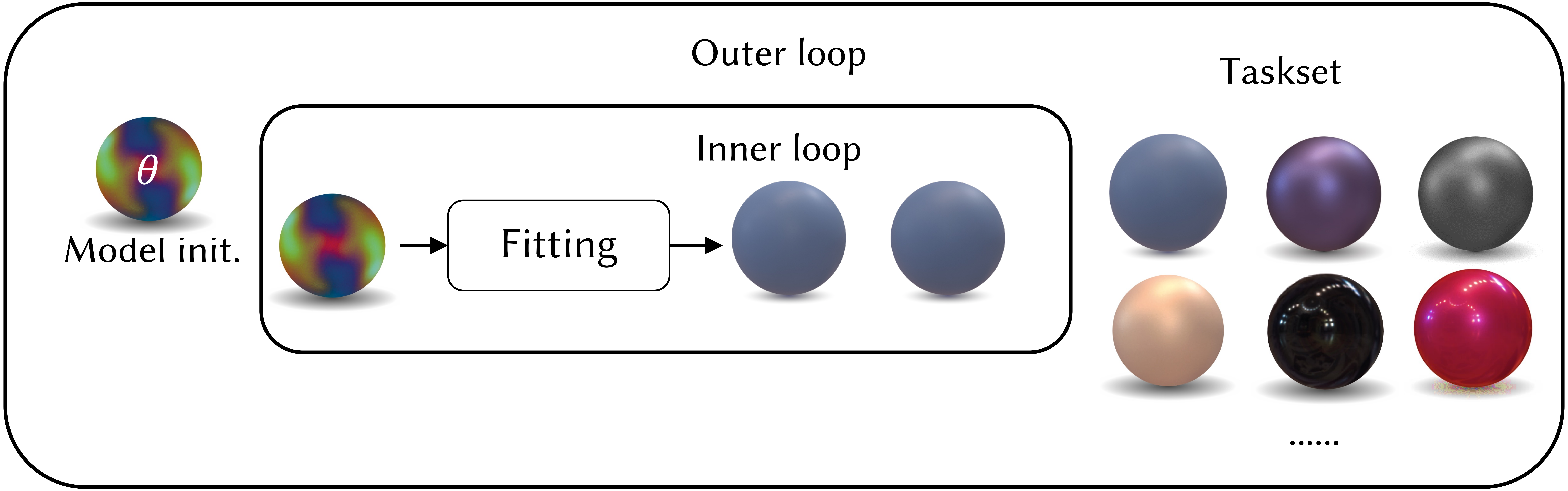


Our solution: meta-sampling

Meta learning in Computer Graphics

- Metappearance [FR22]
- MetaSDF [SCT*20]

- Target parameter: model init.

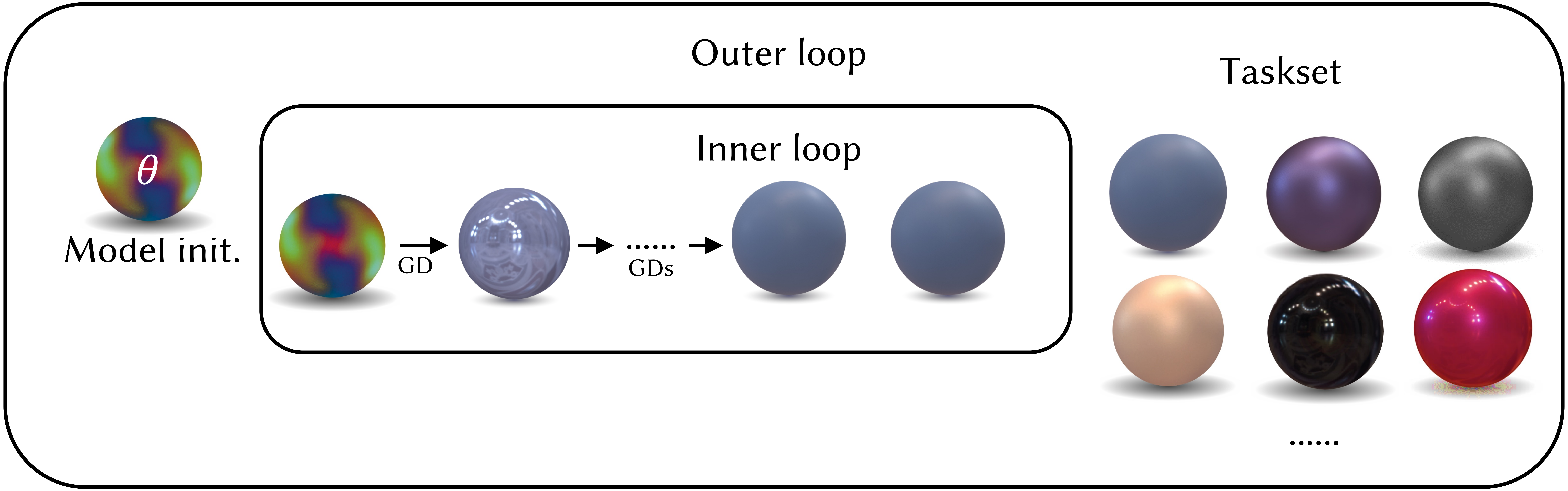


Our solution: meta-sampling

Meta learning in Computer Graphics

- Metappearance [FR22]
- MetaSDF [SCT*20]

- Target parameter: model init.
- Fitting: Stochastic Gradient Descent (SGD)

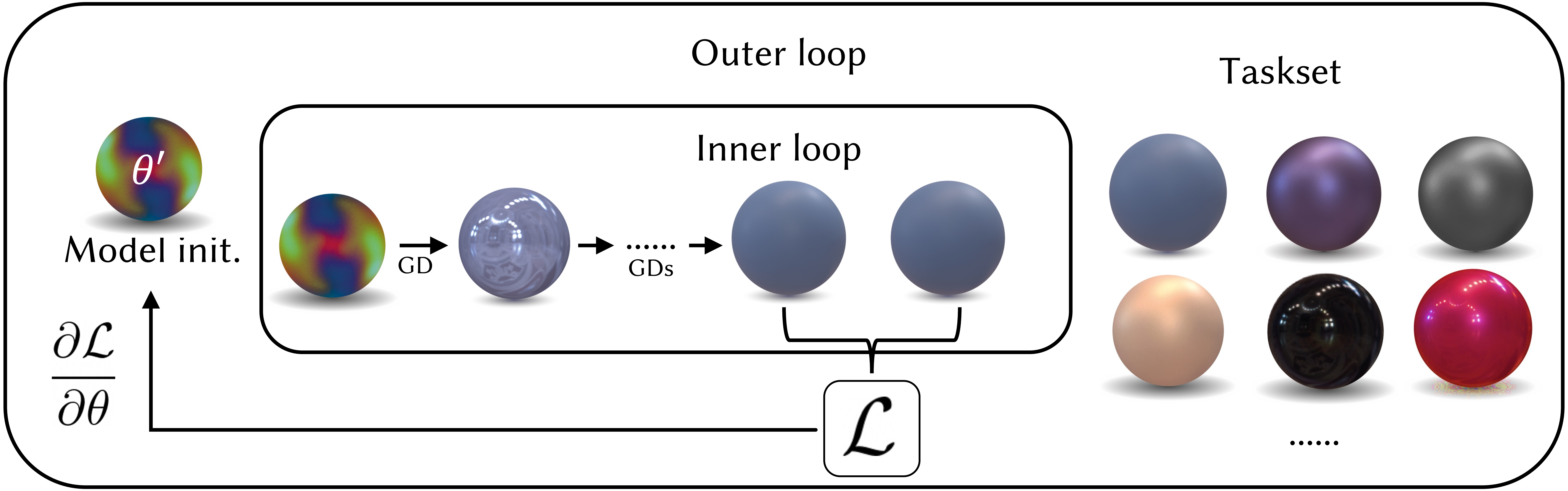


Our solution: meta-sampling

Meta learning in Computer Graphics

- Metappearance [FR22]
- MetaSDF [SCT*20]

- Target parameter: model init.
- Fitting: Stochastic Gradient Descent (SGD)



Our solution: meta-sampling

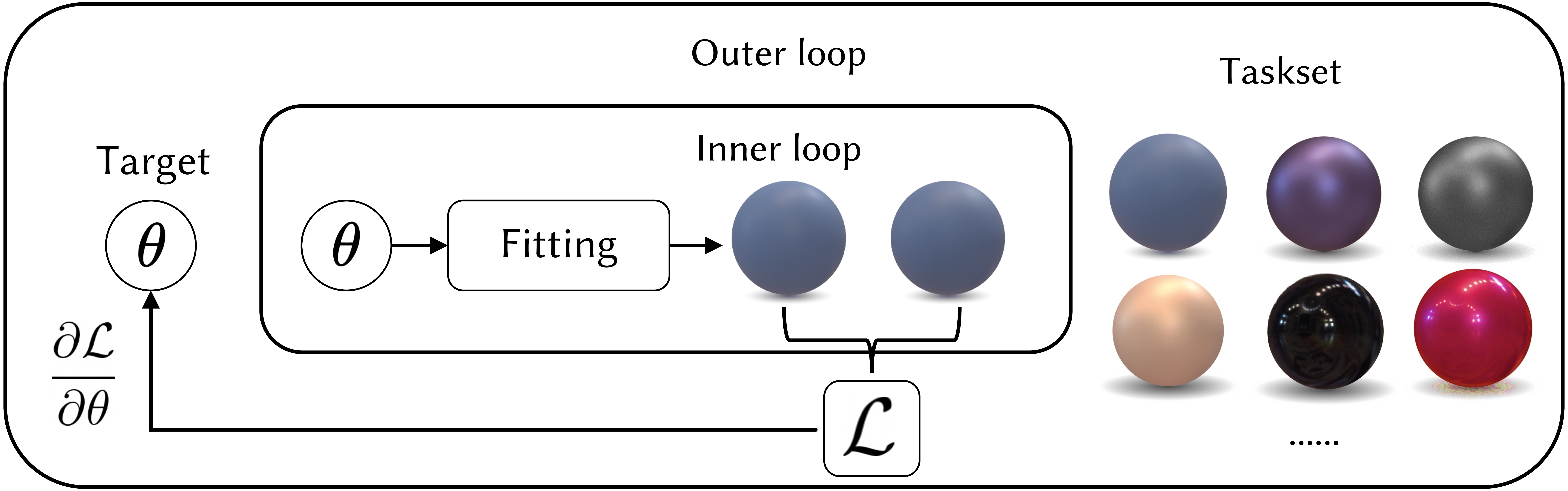
Meta sampling

- the sampler is our target parameter $\boldsymbol{\theta}$

Our solution: meta-sampling

Meta sampling

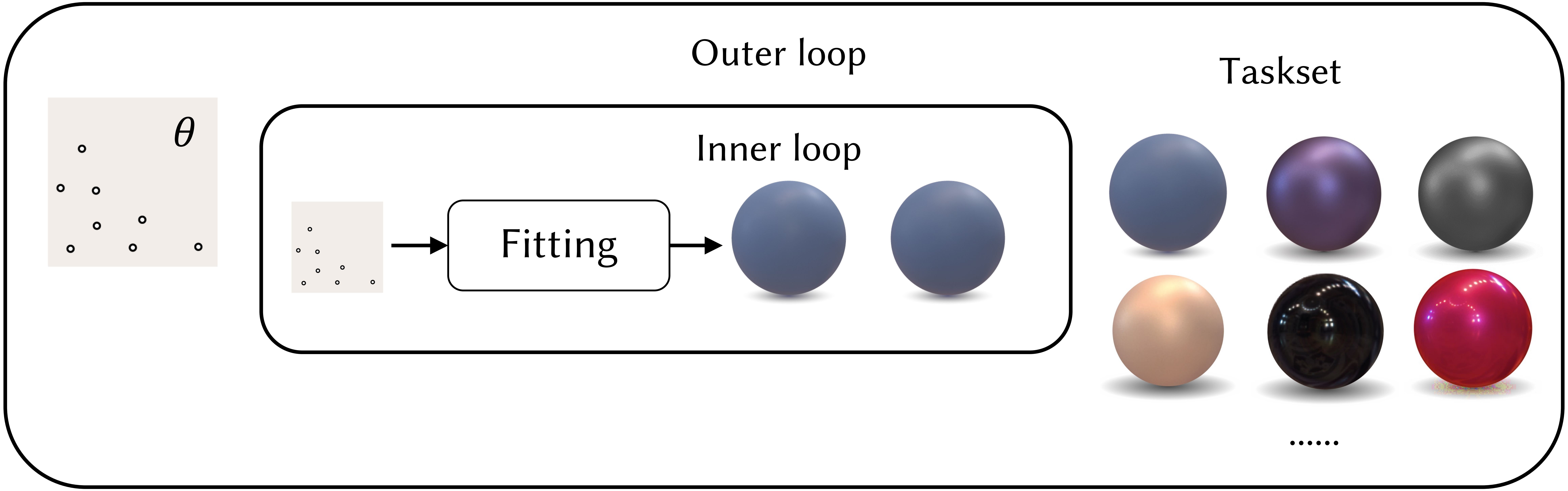
- the sampler is our target parameter $\boldsymbol{\theta}$



Our solution: meta-sampling

Meta sampling

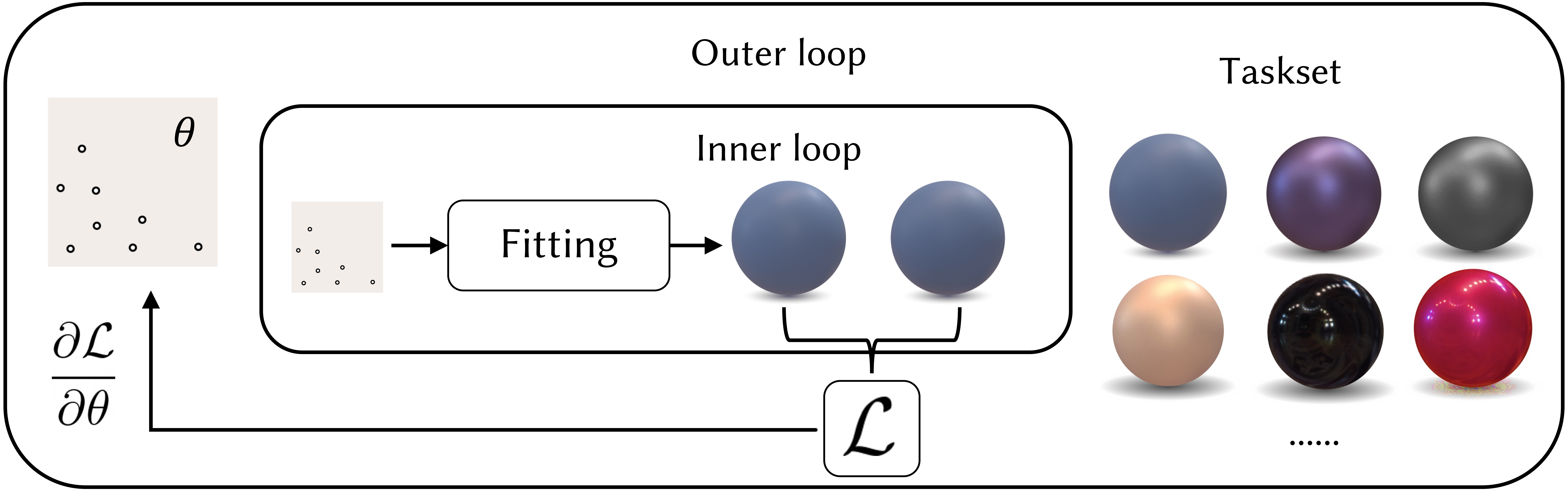
- the sampler is our target parameter $\boldsymbol{\theta}$



Our solution: meta-sampling

Meta sampling

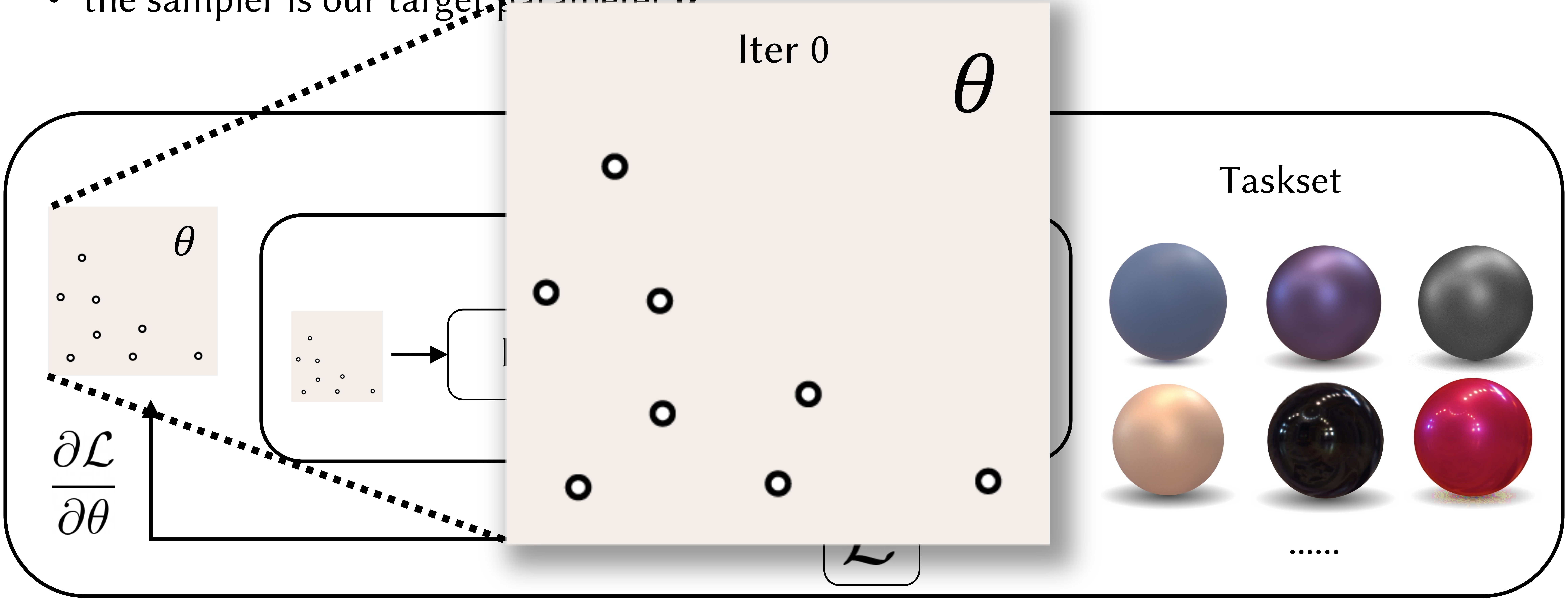
- the sampler is our target parameter $\boldsymbol{\theta}$



Our solution: meta-sampling

Meta sampling

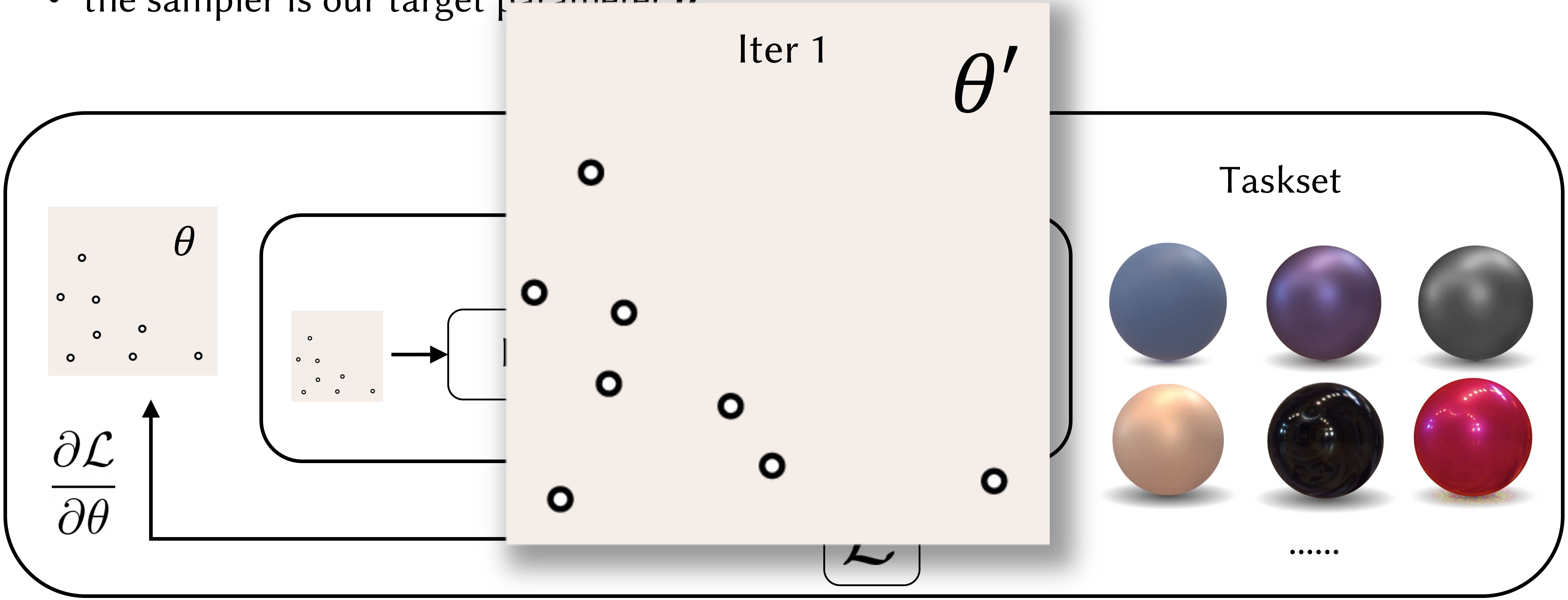
- the sampler is our target parameter $\boldsymbol{\theta}$



Our solution: meta-sampling

Meta sampling

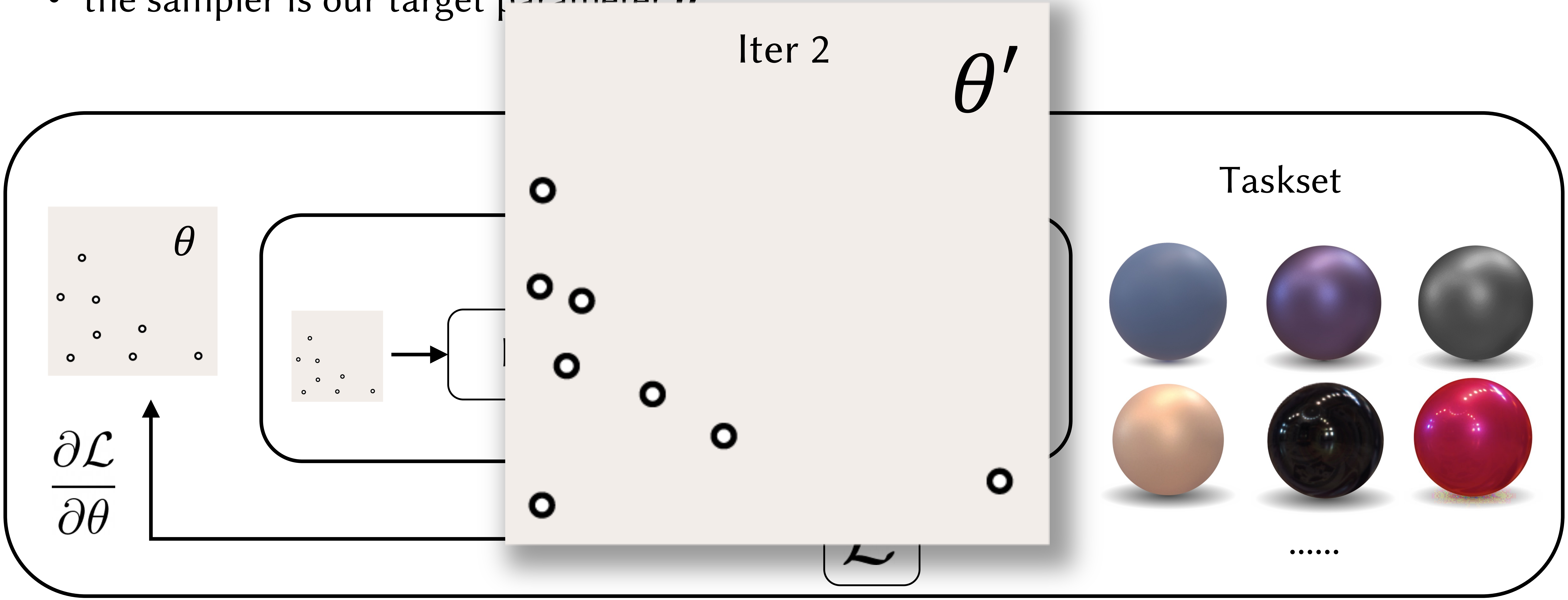
- the sampler is our target parameter $\boldsymbol{\theta}$



Our solution: meta-sampling

Meta sampling

- the sampler is our target parameter $\boldsymbol{\theta}$

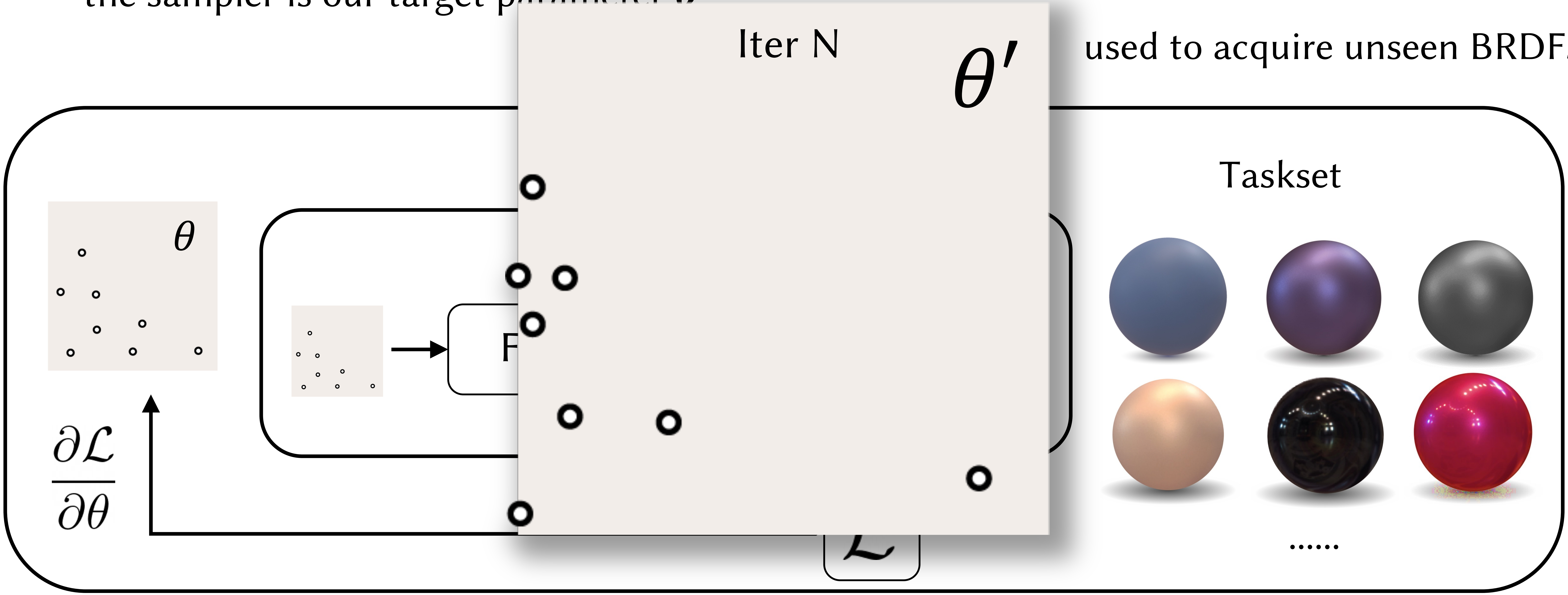


Our solution: meta-sampling

Meta sampling

- the sampler is our target parameter $\boldsymbol{\theta}$

used to acquire unseen BRDFs



For non-linear BRDF models

Using SGD in inner loop

- too many steps are prohibitive by cost
- 20 steps are not enough to fully make use of samples
- Use a meta-learned initialization [FR22]

Results

Optimization

Sampling

Results

BRDF 1

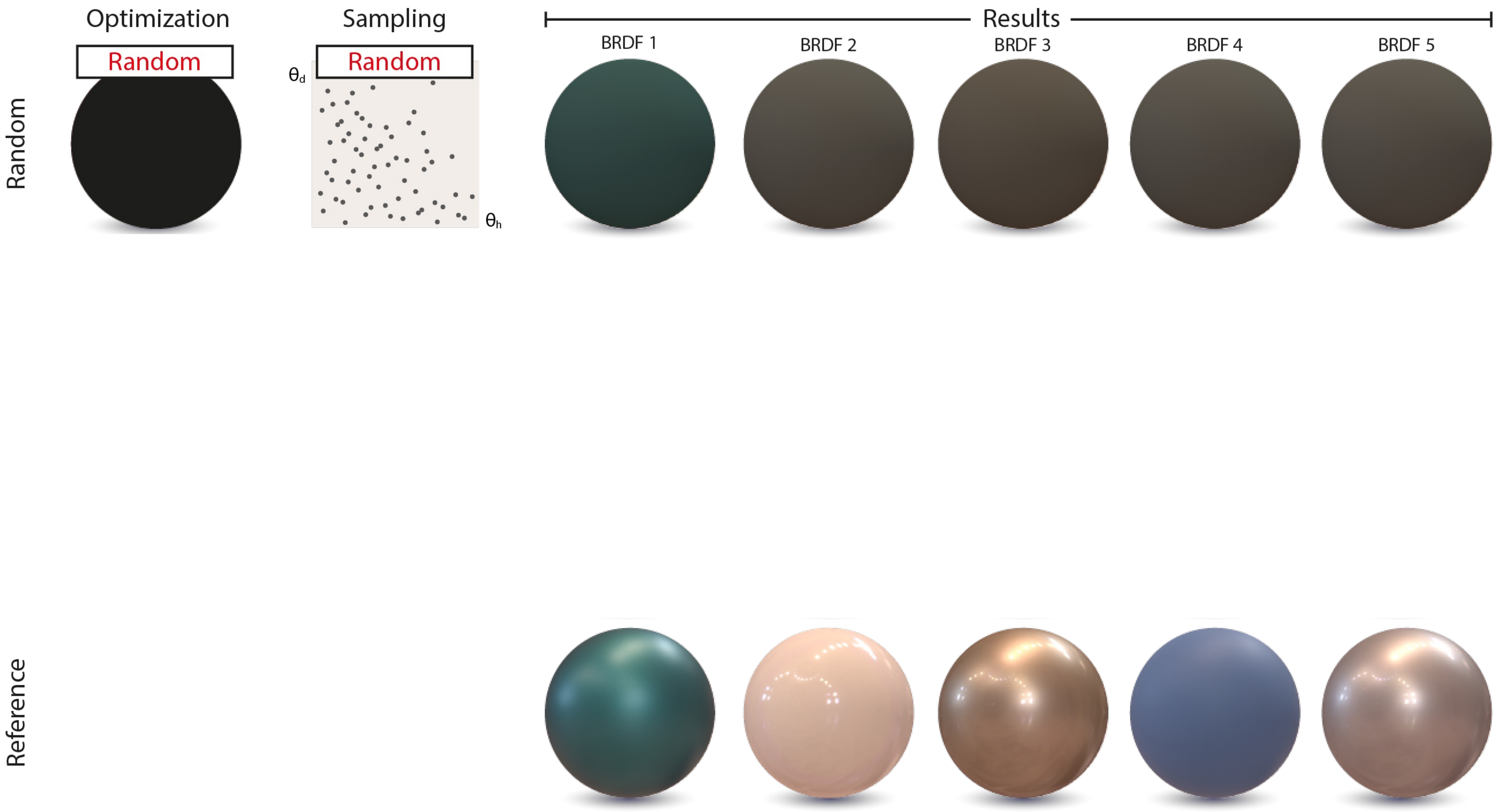
BRDF 2

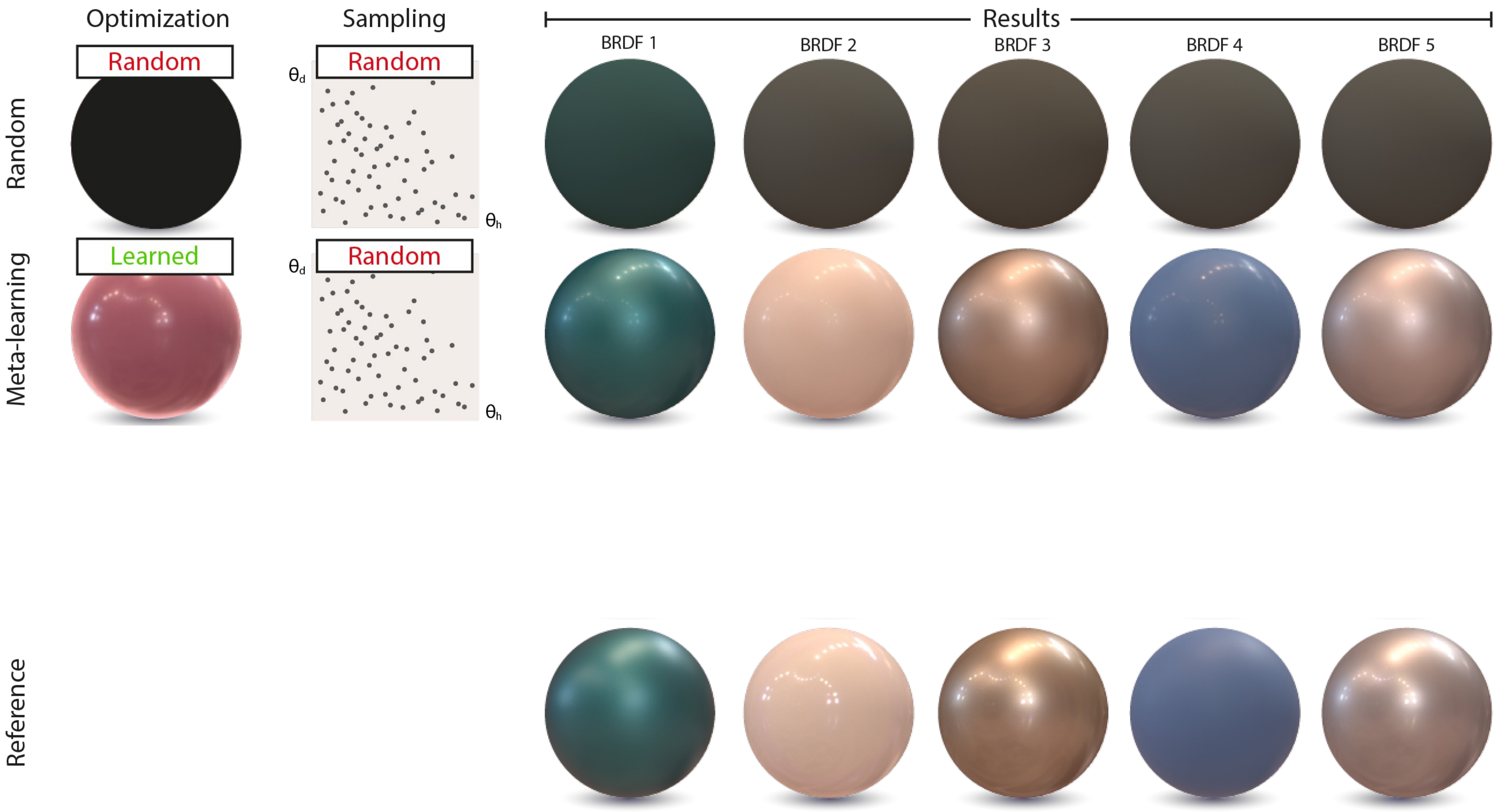
BRDF 3

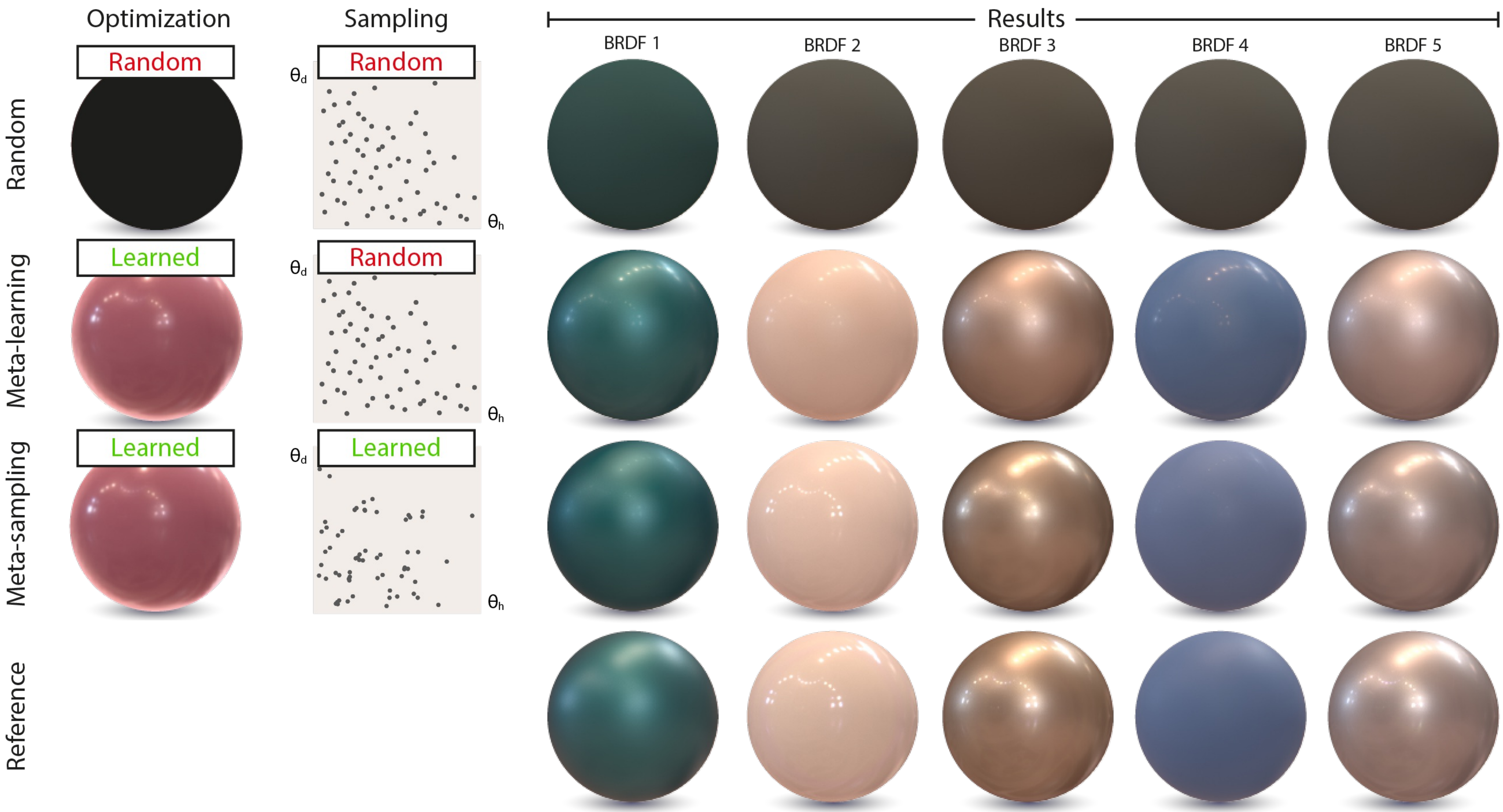
BRDF 4

BRDF 5

Reference


Optimization
Sampling
Results
BRDF 1
BRDF 2
BRDF 3
BRDF 4
BRDF 5
Random
Random
Random
θ_d
θ_h
Reference

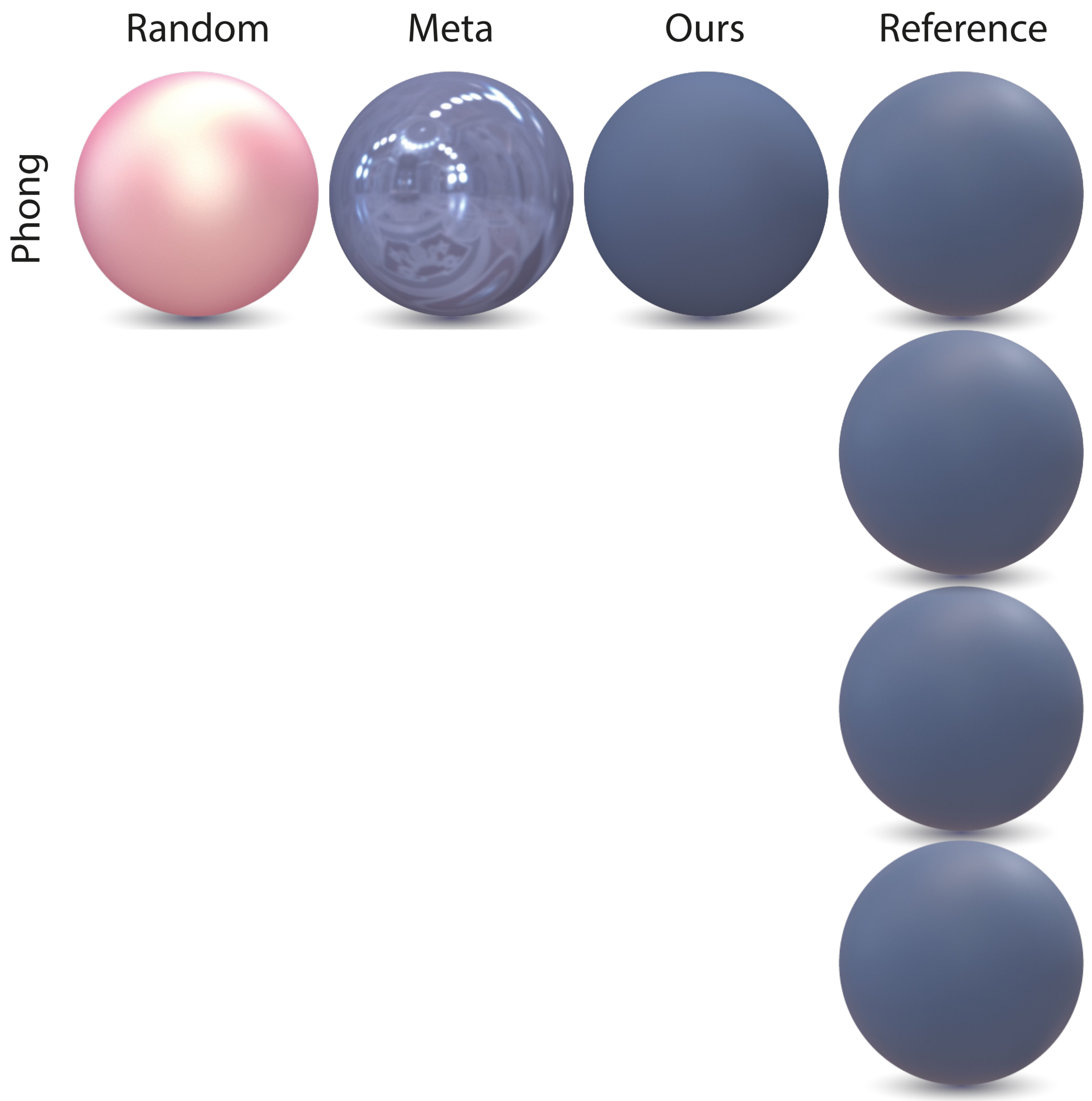

Optimization
Sampling
Results
BRDF 1
BRDF 2
BRDF 3
BRDF 4
BRDF 5
Random
Random
Random
θd
θh
Meta-learning
Learned
Random
θd
θh
Reference


Optimization
Sampling
Results
BRDF 1
BRDF 2
BRDF 3
BRDF 4
BRDF 5
Random
Meta-learning
Meta-sampling
Reference
Random
Random
Learned
Random
Learned
Learned
θ_d
θ_h
θ_d
θ_h
θ_d
θ_h

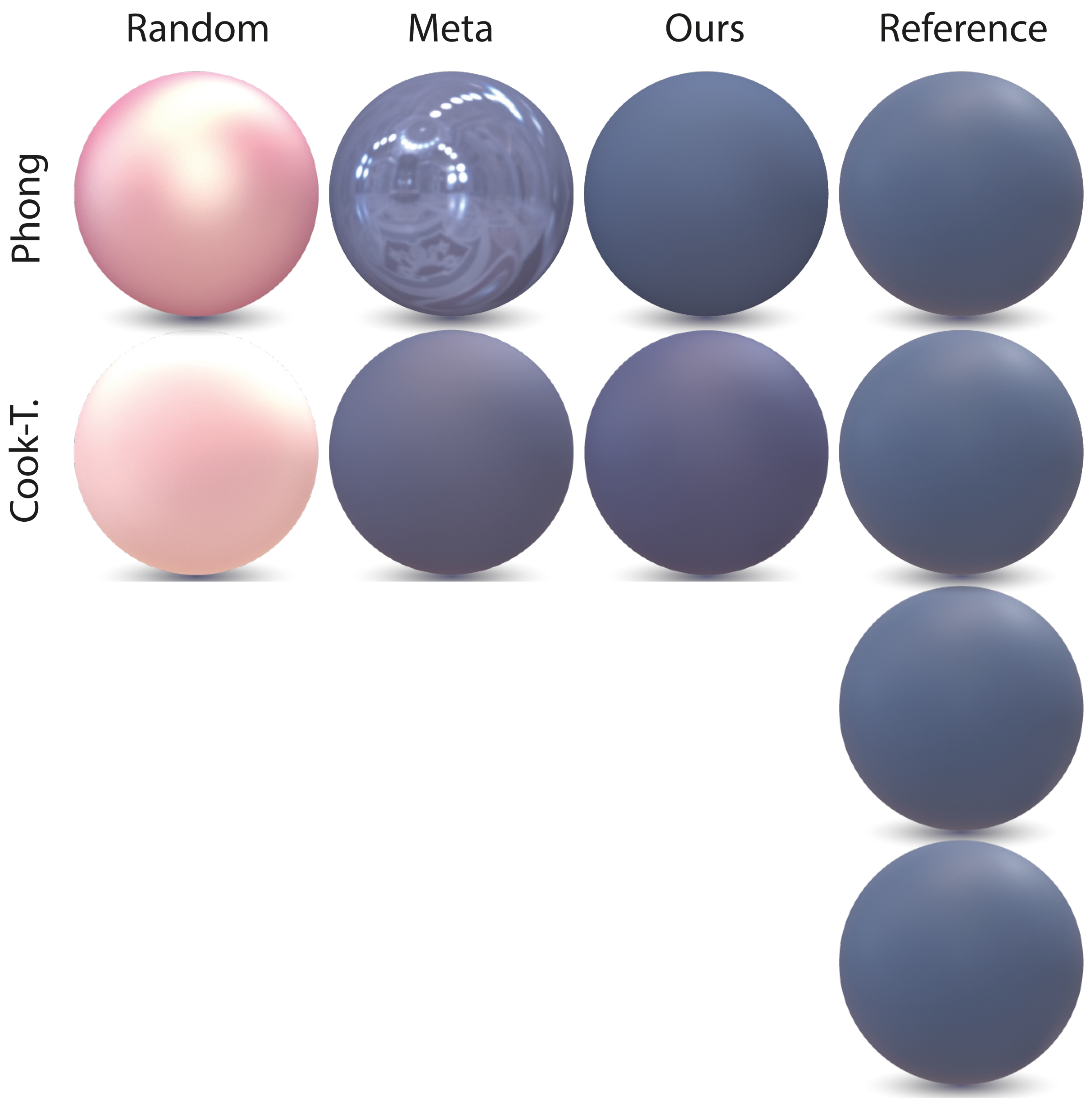
#samples = 8,
for the diffuse BRDF

Random | Meta | Ours | Reference

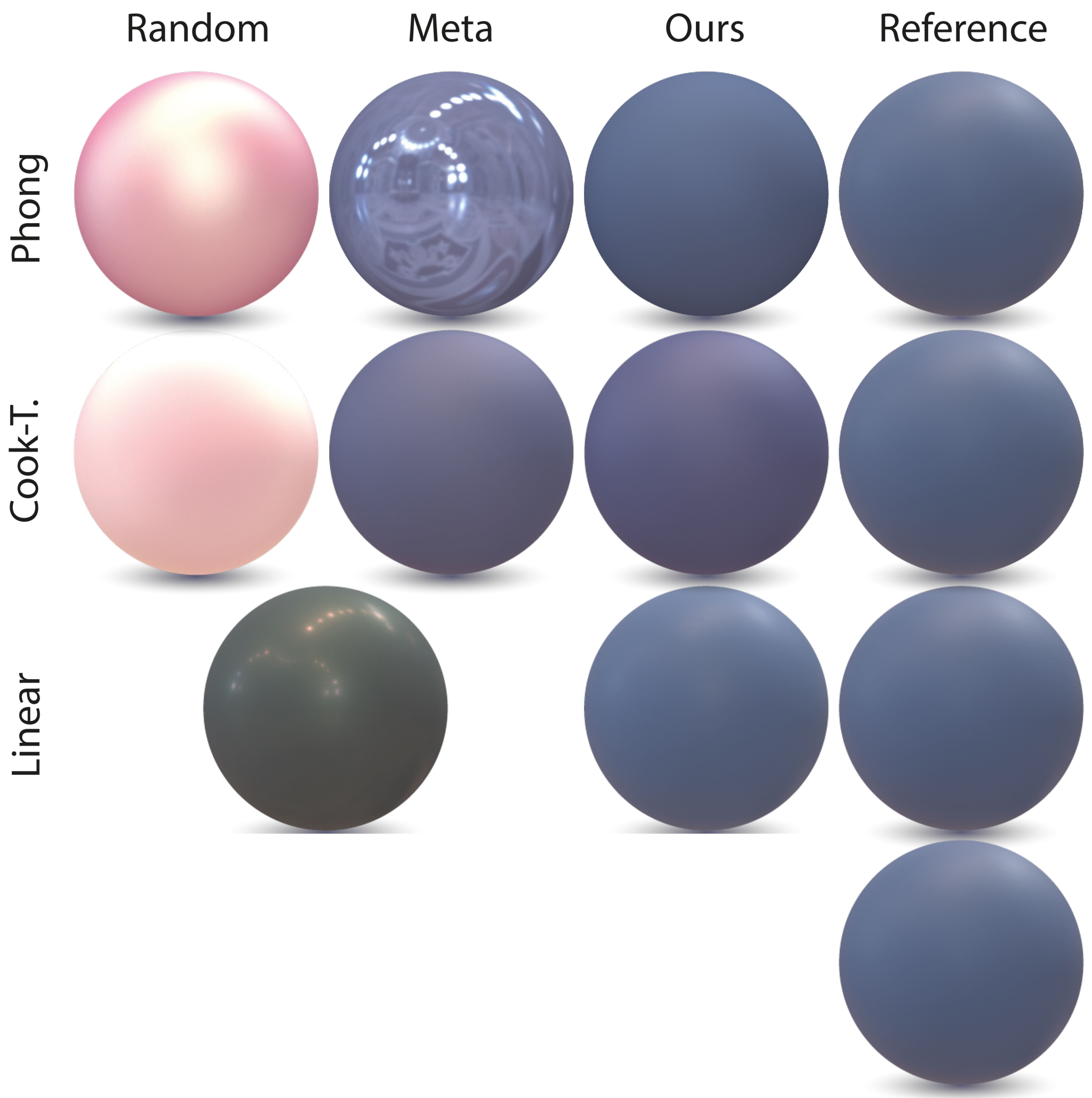
#samples = 8,
for the diffuse BRDF



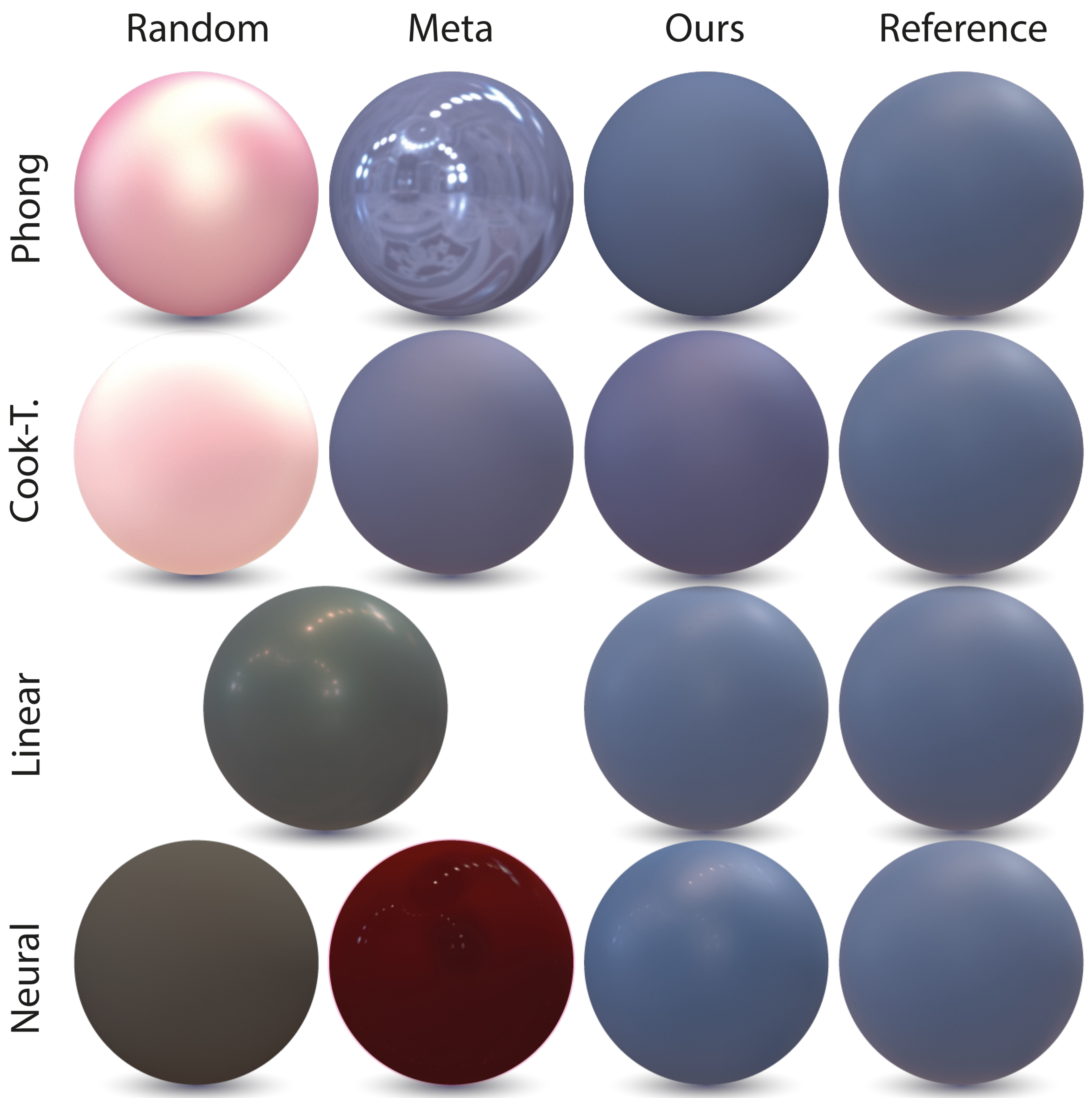
#samples = 8,
for the diffuse BRDF



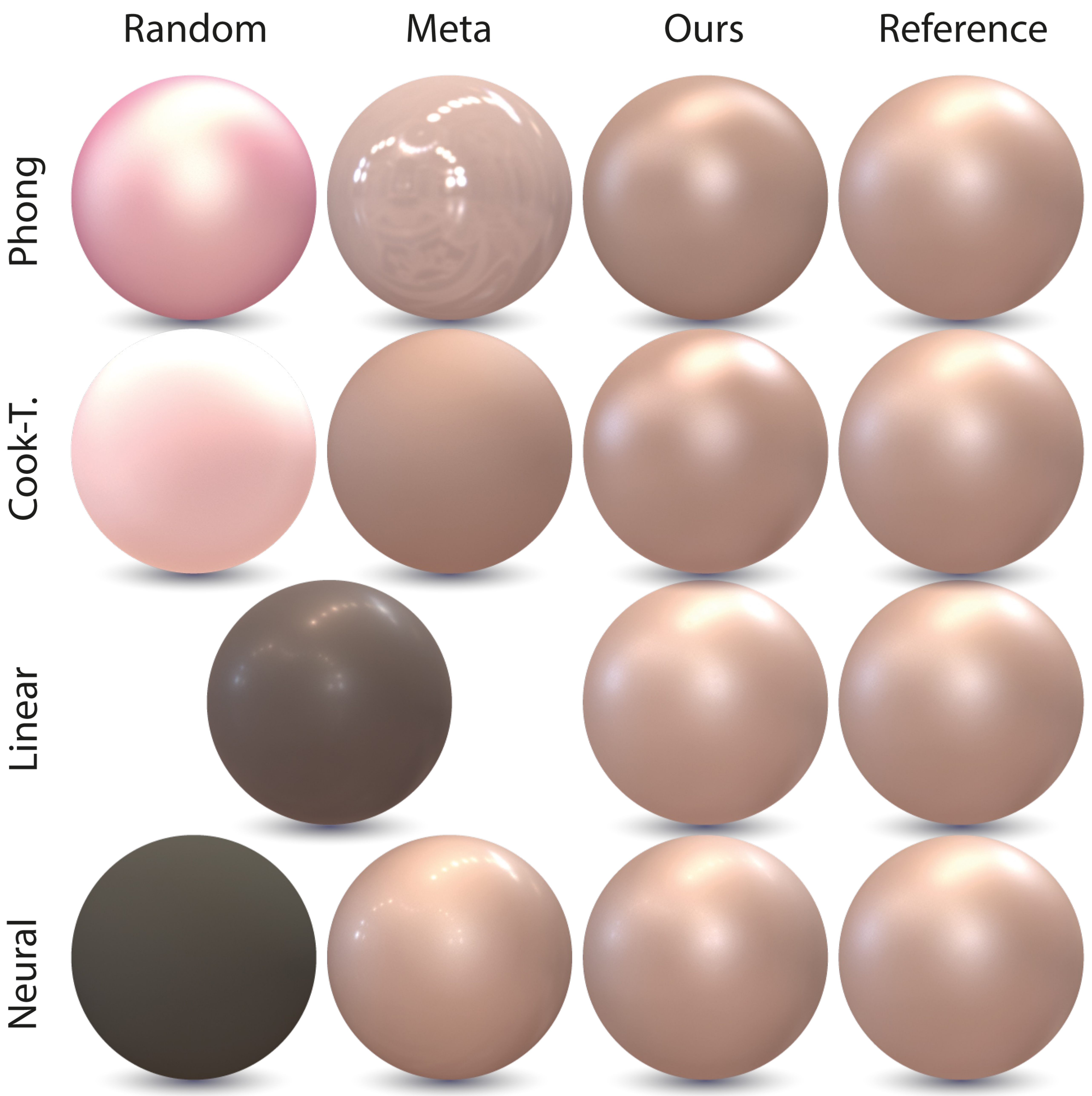
#samples = 8,
for the diffuse BRDF



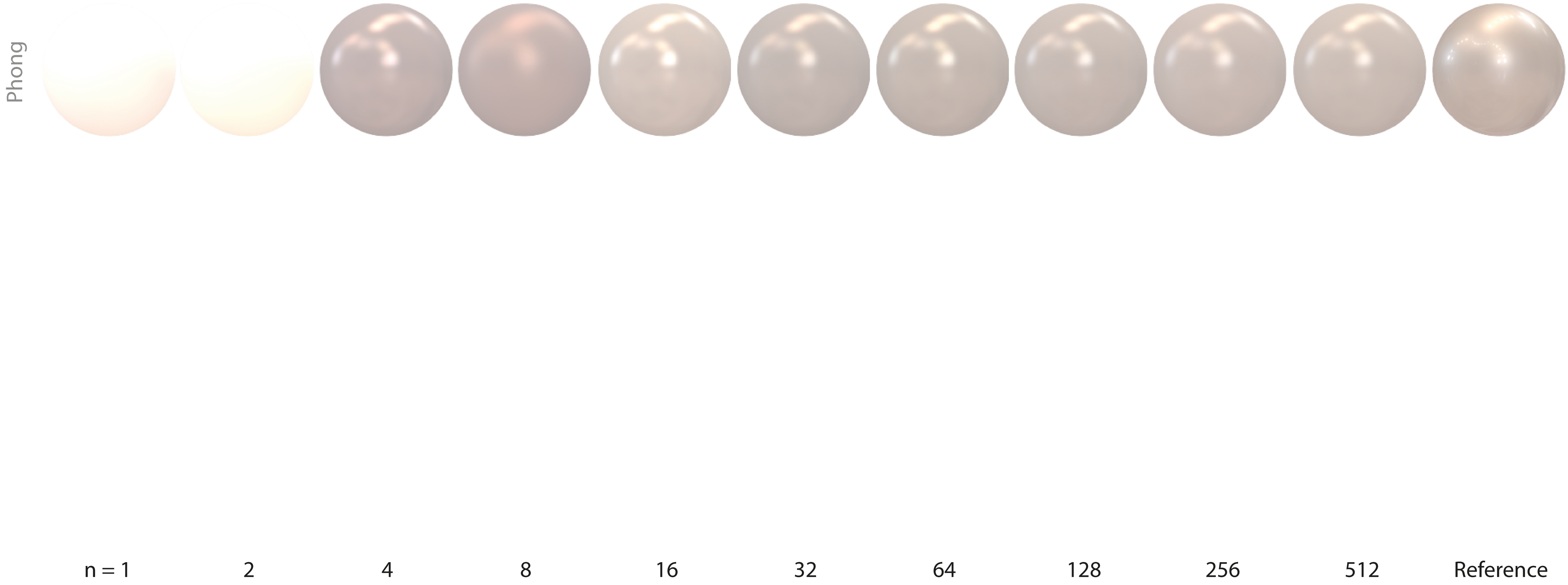
#samples = 8,
for the diffuse BRDF



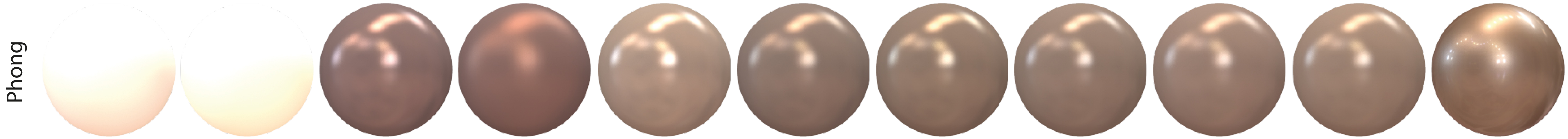
#samples = 32, for the glossy BRDF



Increasing #samples

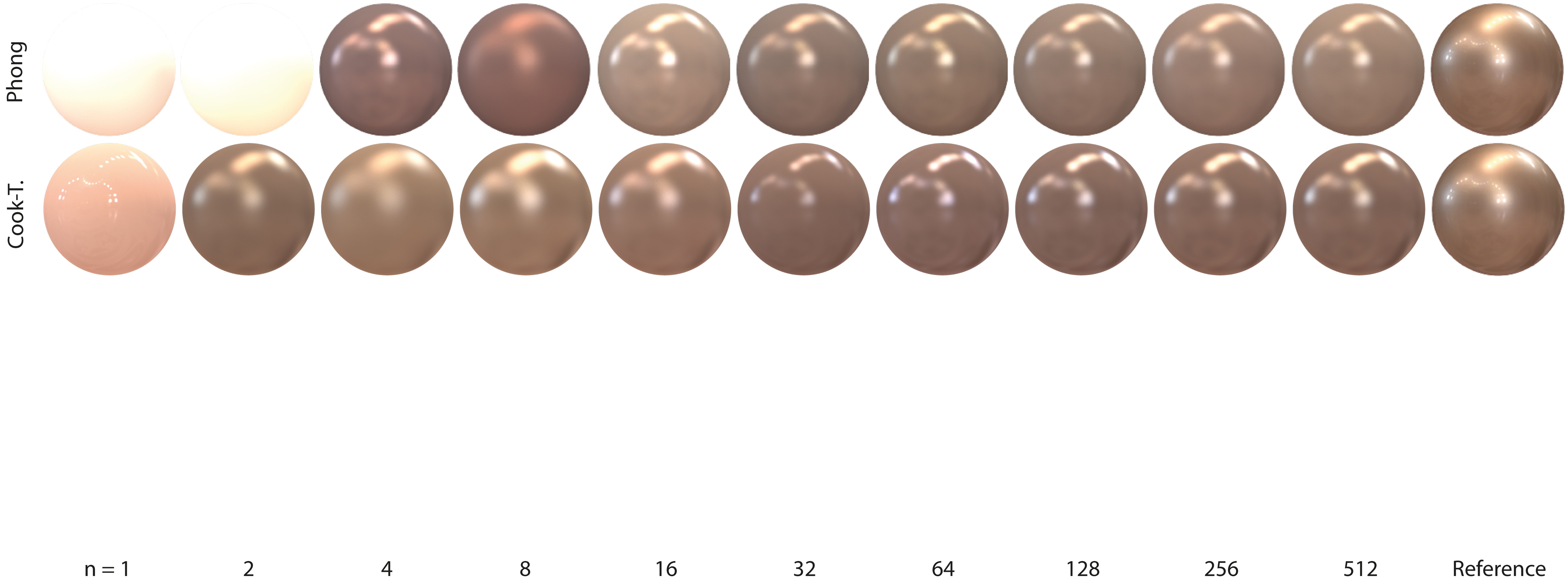


Increasing #samples

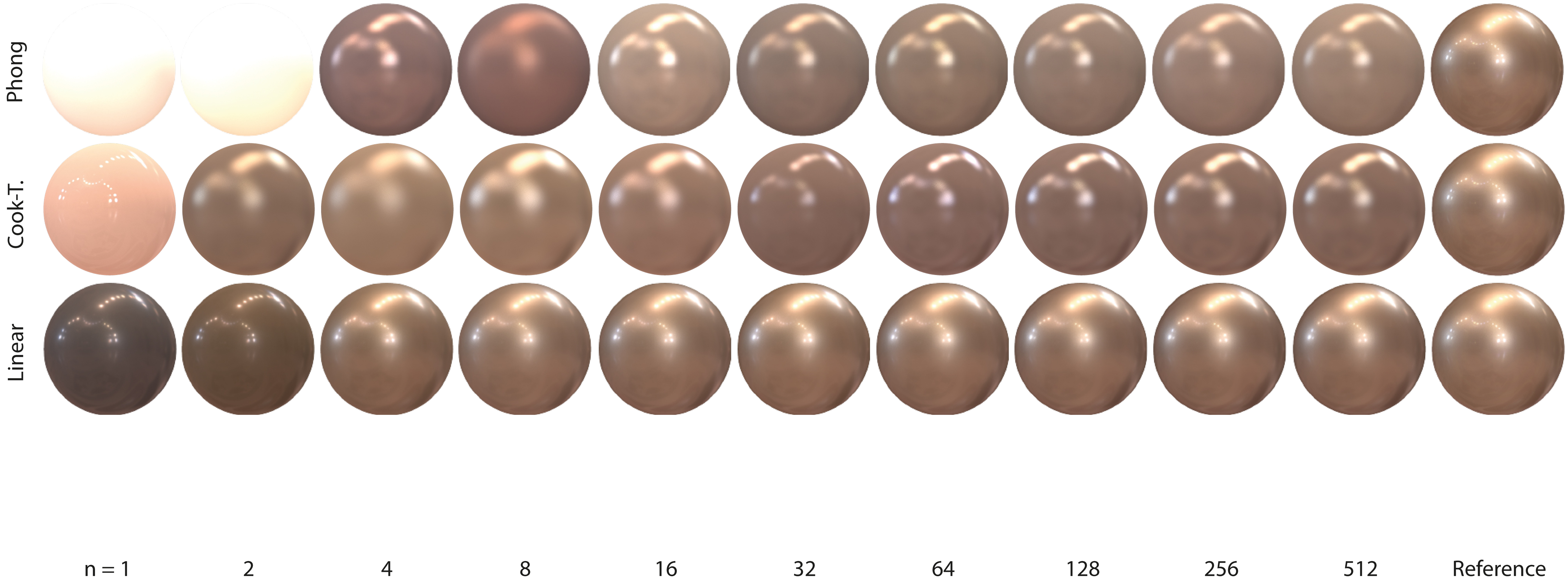


n = 1 | 2 | 4 | 8 | 16 | 32 | 64 | 128 | 256 | 512 | Reference

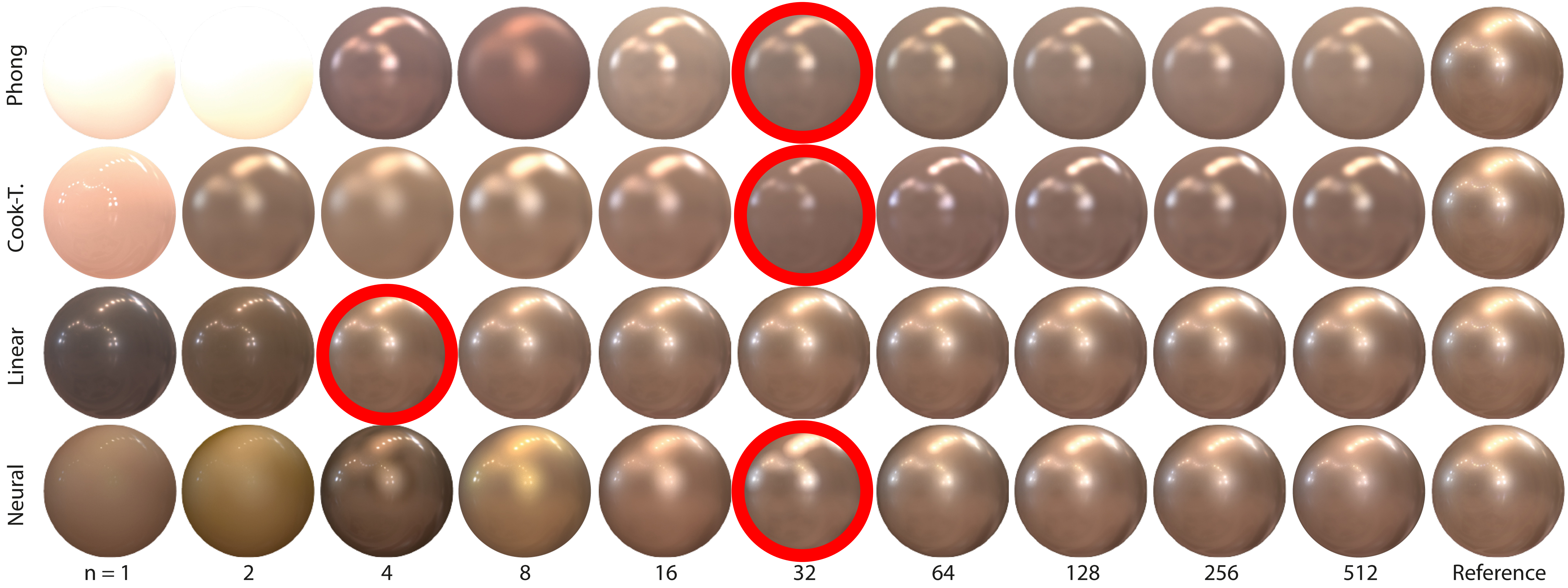
Increasing #samples



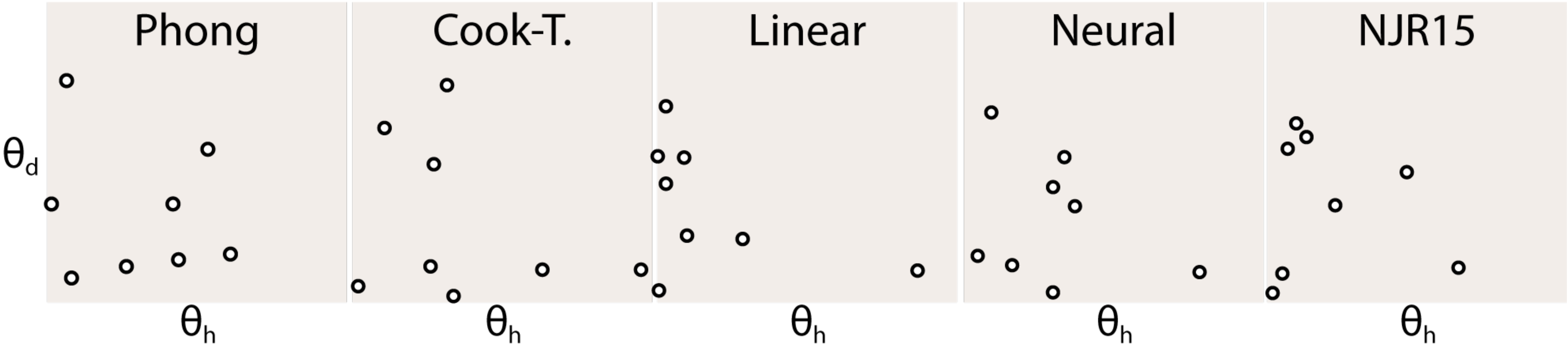
Increasing #samples



Increasing #samples



Learned patterns



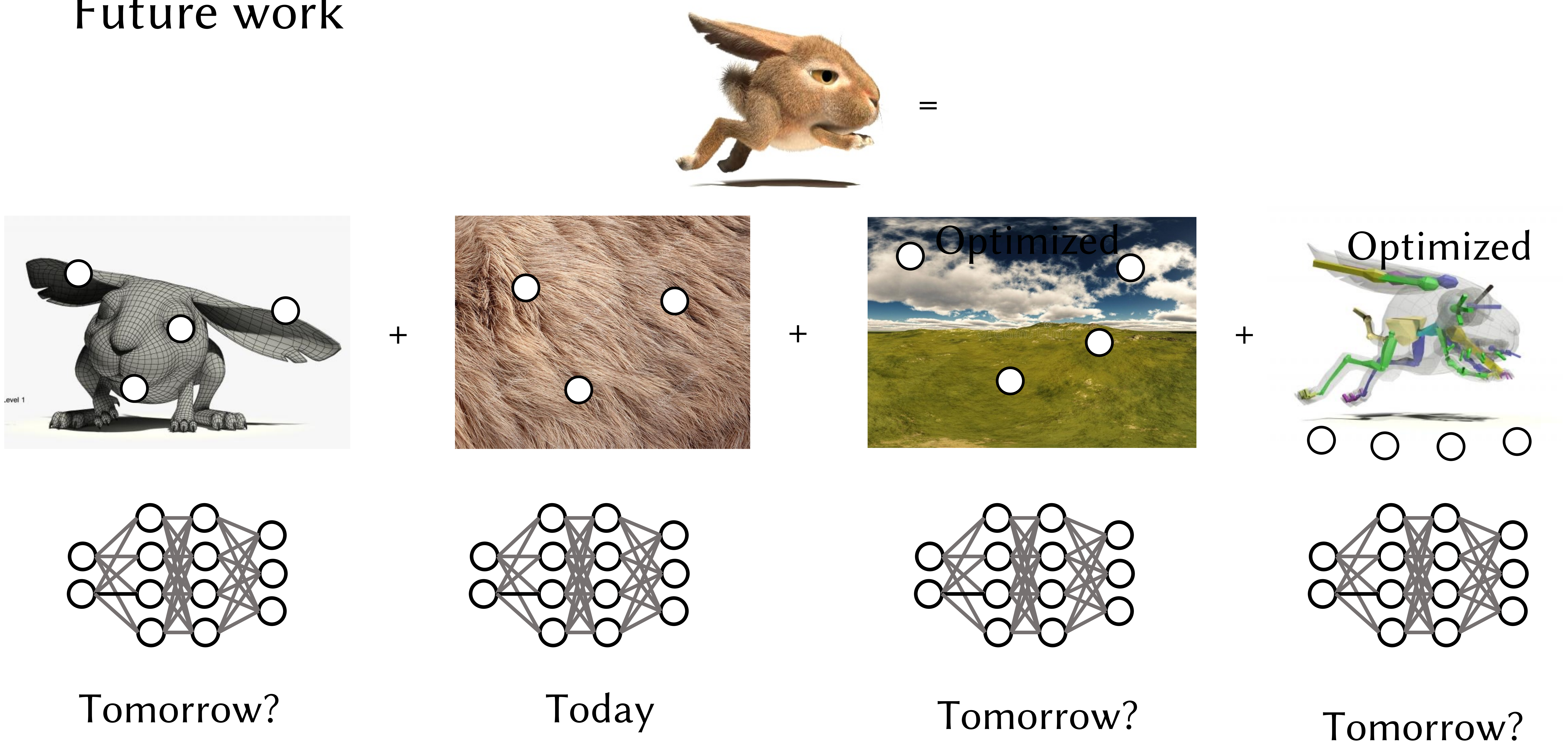
Still difficult to interpret these patterns

Summary

Meta sampling

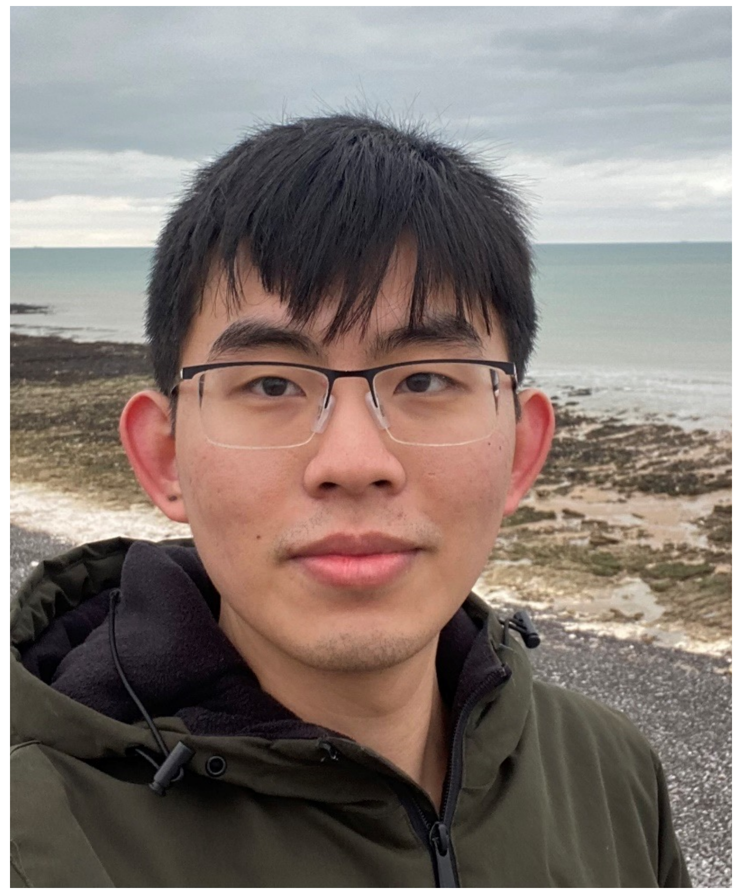
- Model-agnostic: Neural, Linear, and Analytical… all good!
- Performance
  - Reconstruct high-quality BRDFs by only 4~32 samples
  - 5 orders of magnitutdes fewer
- Compared to [NJR15]
  - Extended to more BRDF models
  - Better loss using same number of samples

Future work



Meta-sampling: Learning to Learn and Sample BRDFs

ryushinn.github.io/metasampling



Chen Liu

Michael Fischer

Tobias Ritschel

We acknowledged funding by Meta Reality Labs

References

[PB75] Illumination for Computer Generated Pictures

[CT82] A Reflectance Model for Computer Graphics

[Mat03] A Data-driven Reflectance Model

[NJR15] On Optimal, Minimal BRDF Sampling for Reflectance Acquisition

[SRRW21] Neural BRDF Representation and Importance Sampling

[FR22] Metappearance: Meta-Learning for Visual Appearance Reproduction

[SCT*20] MetaSDF: Meta-learning Signed Distance Functions

Thank you!