Introduction to Computational Design
Part 2: Human in the Loop

Presenter: Yuki Koyama (AIST, Japan)

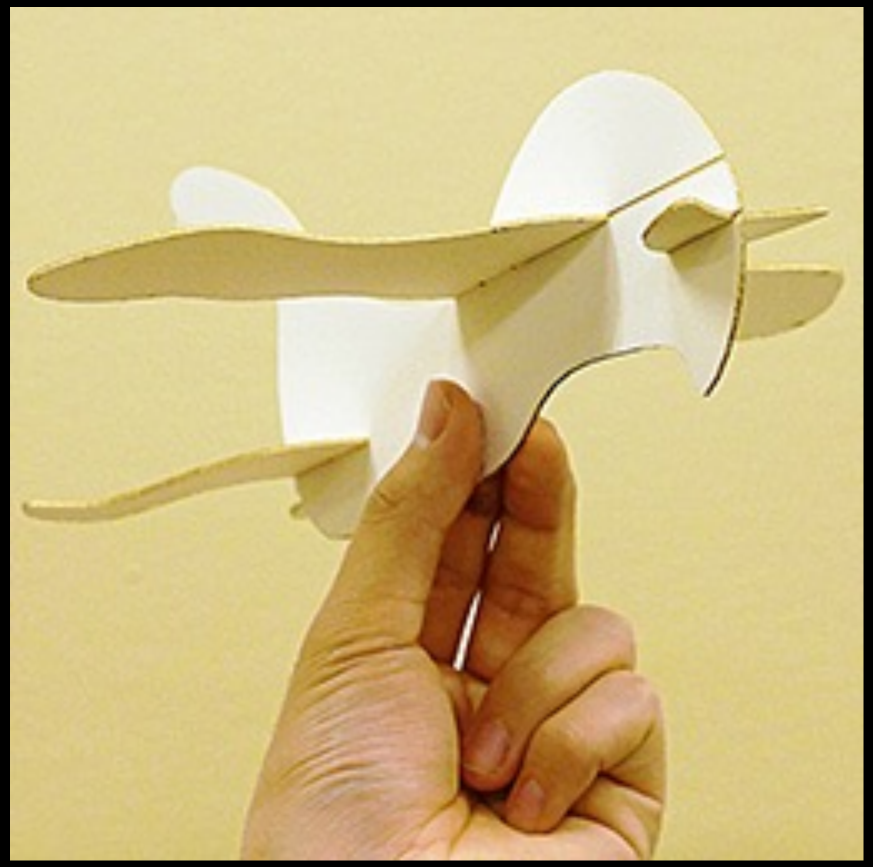
2021-05-12 | CHI 2021 Course | Online

Human-in-the-Loop Optimization: Motivation

Computational design:

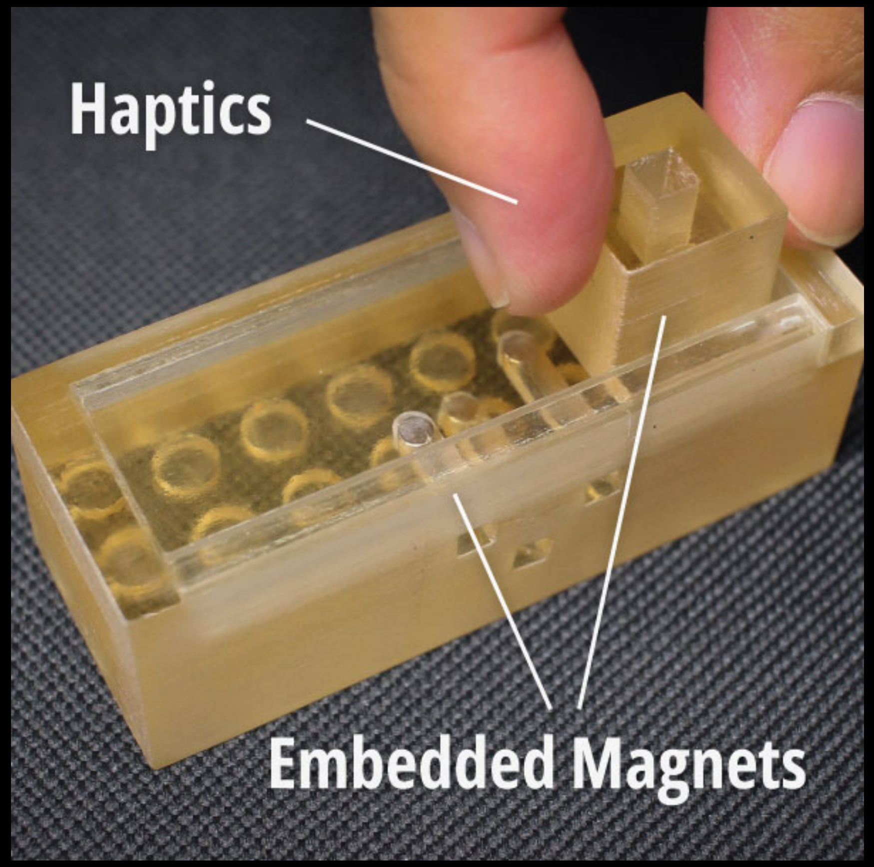
Computational design:

SIGGRAPH 2014



Fly-ability

CHI 2021



Haptics

Pacific Graphics 2016

Fly-ability

SIGGRAPH Asia 2015

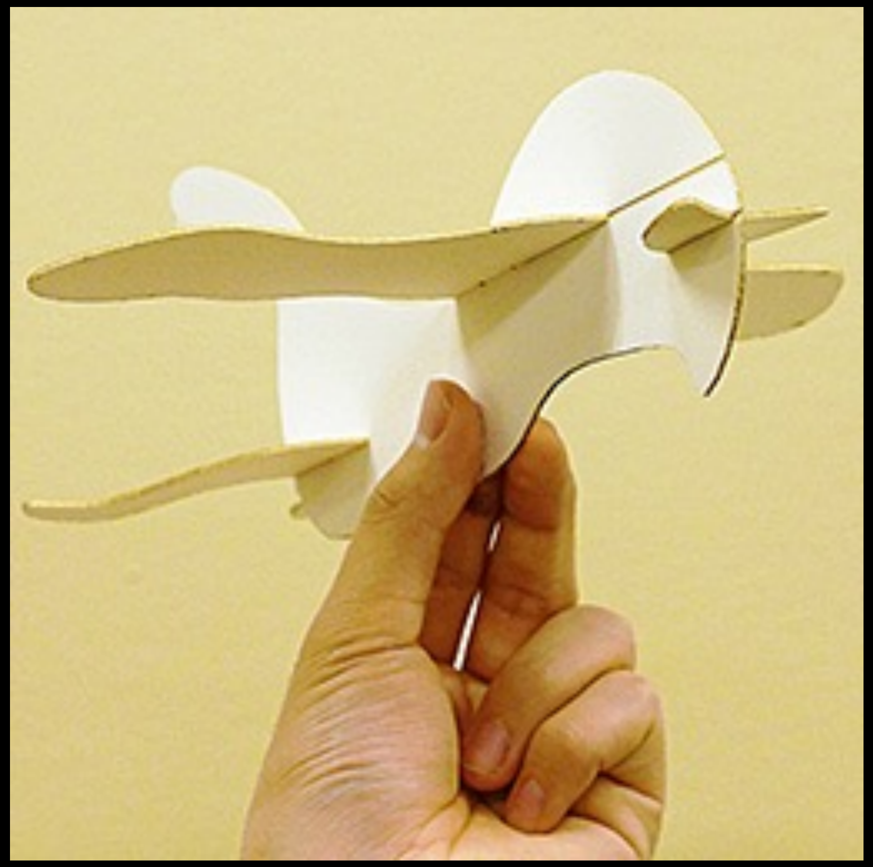
Connect-ability

VRST 2017

Haptics

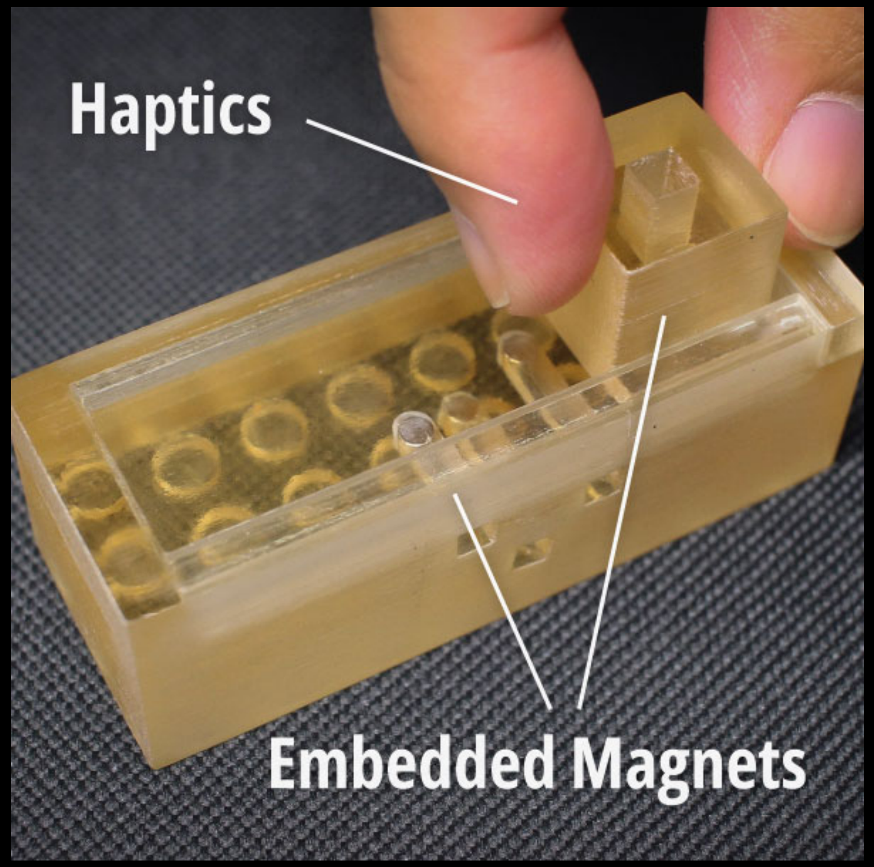
Computational design:

SIGGRAPH 2014



Fly-ability

CHI 2021



Haptics

Pacific Graphics 2016

Fly-ability

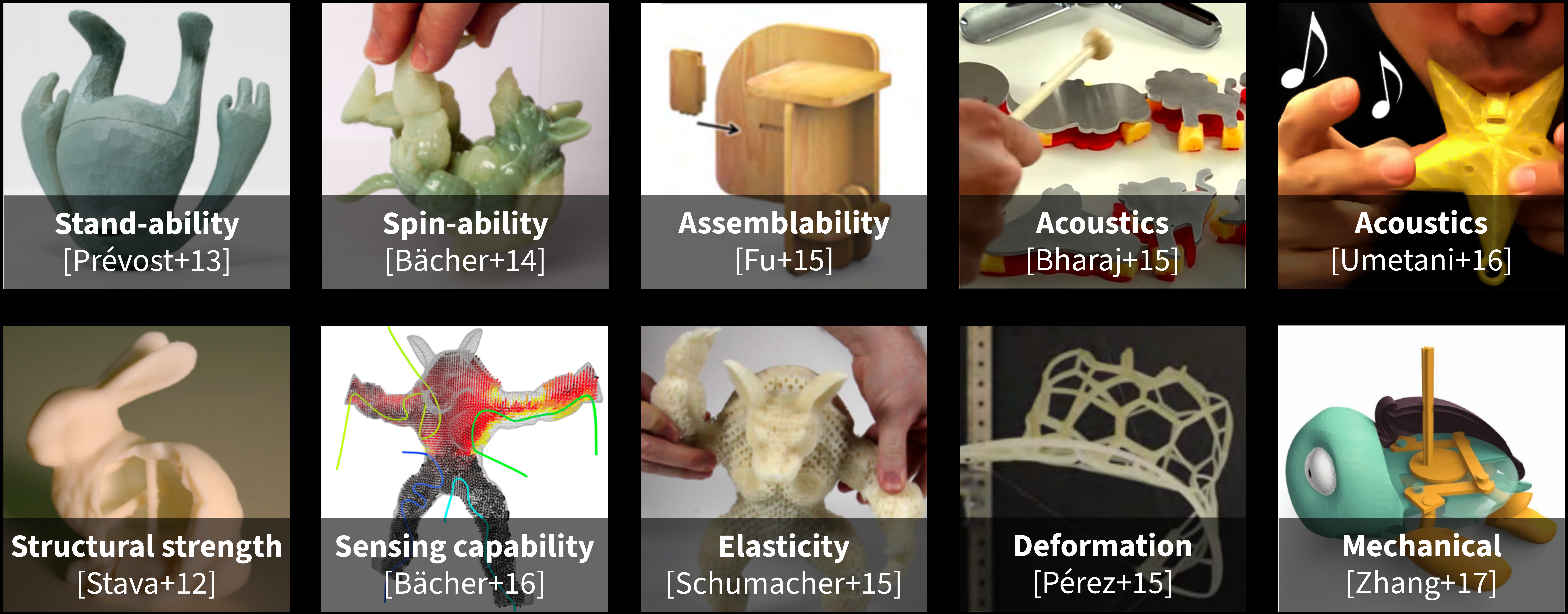
SIGGRAPH Asia 2015

Connect-ability

VRST 2017

Haptics

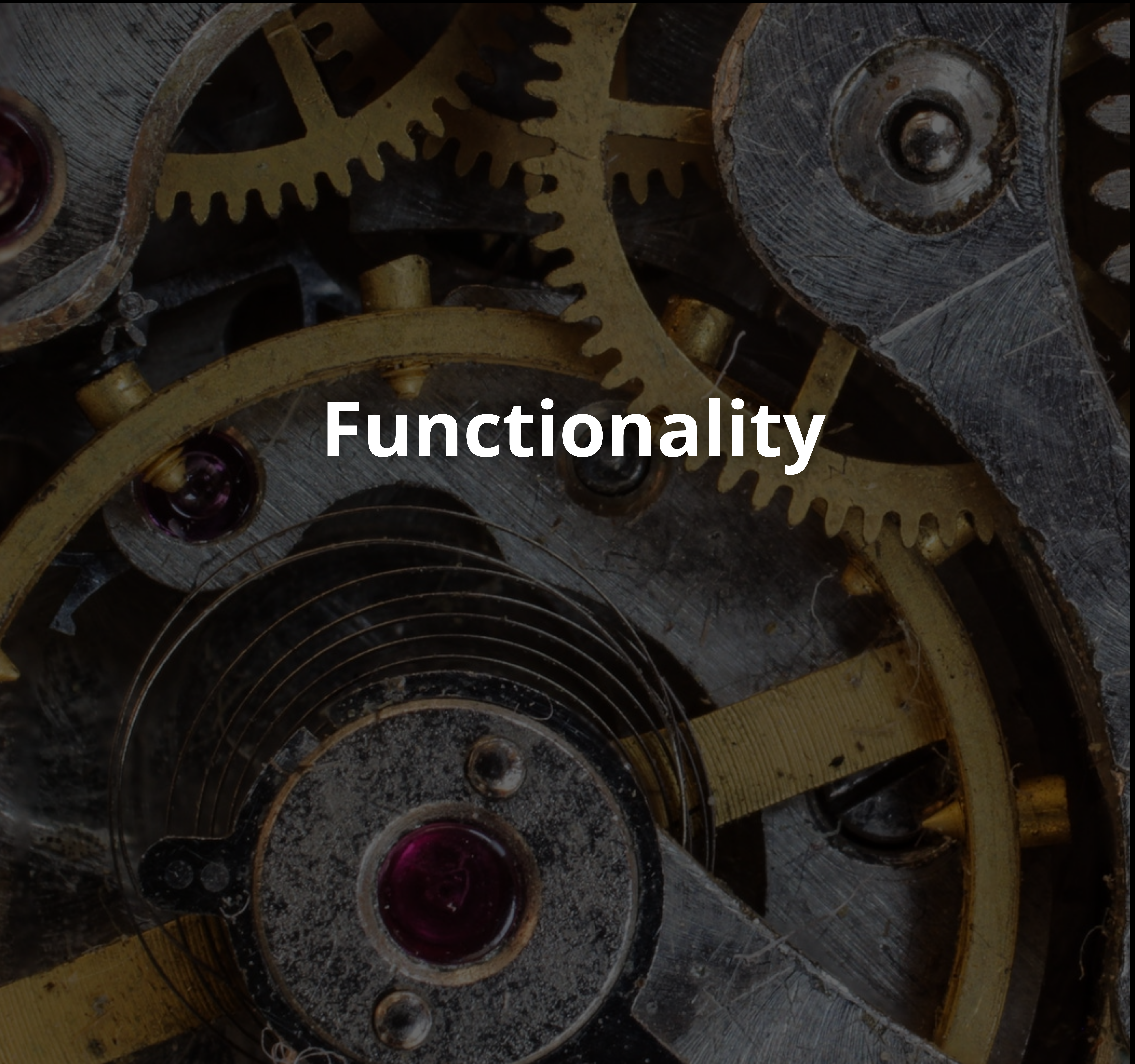
The design goal (objective) is the functionality of the designed object



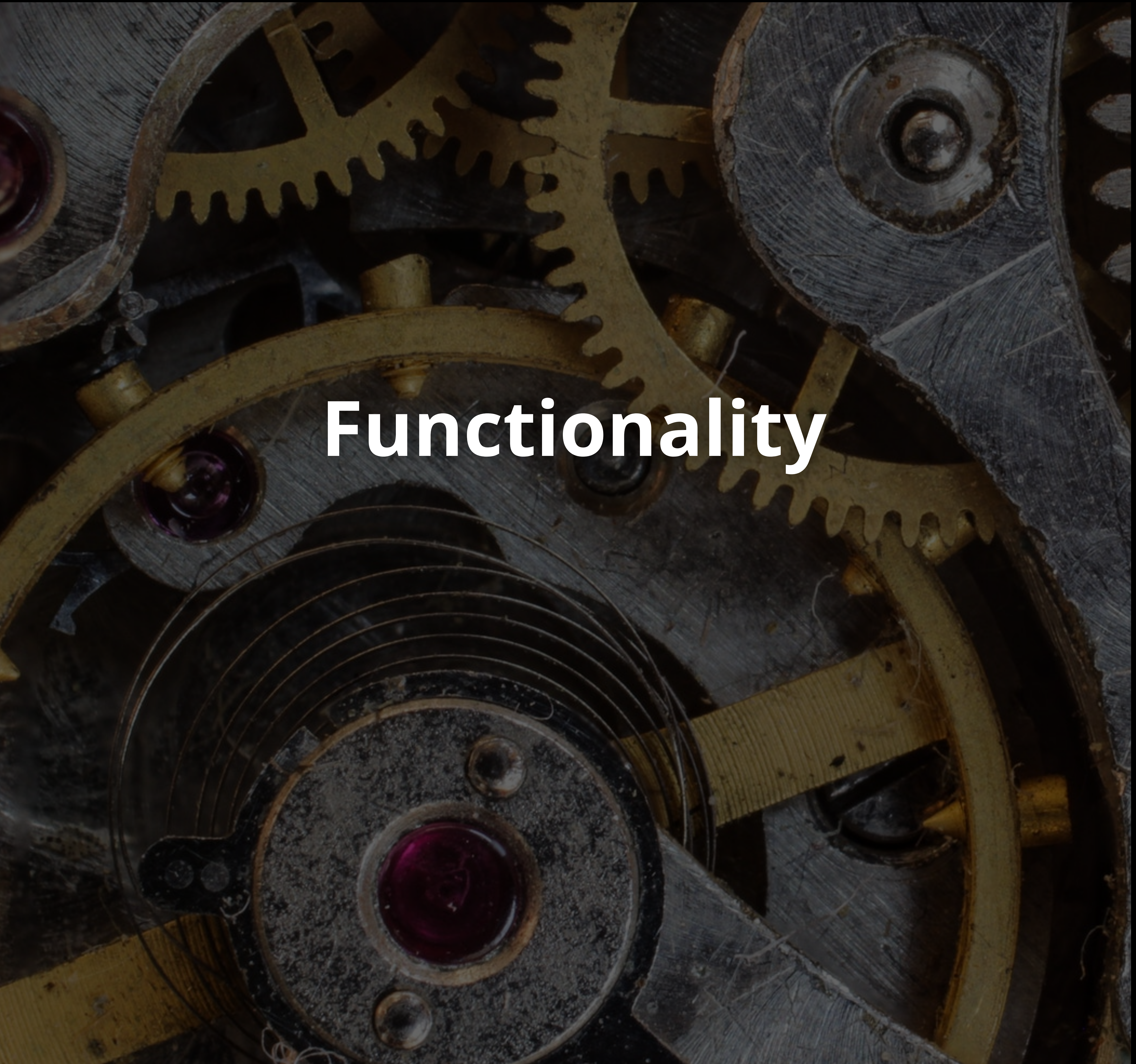
The design goal (objective) is the functionality of the designed object

Design Goals (Objectives)

Design Goals (Objectives)



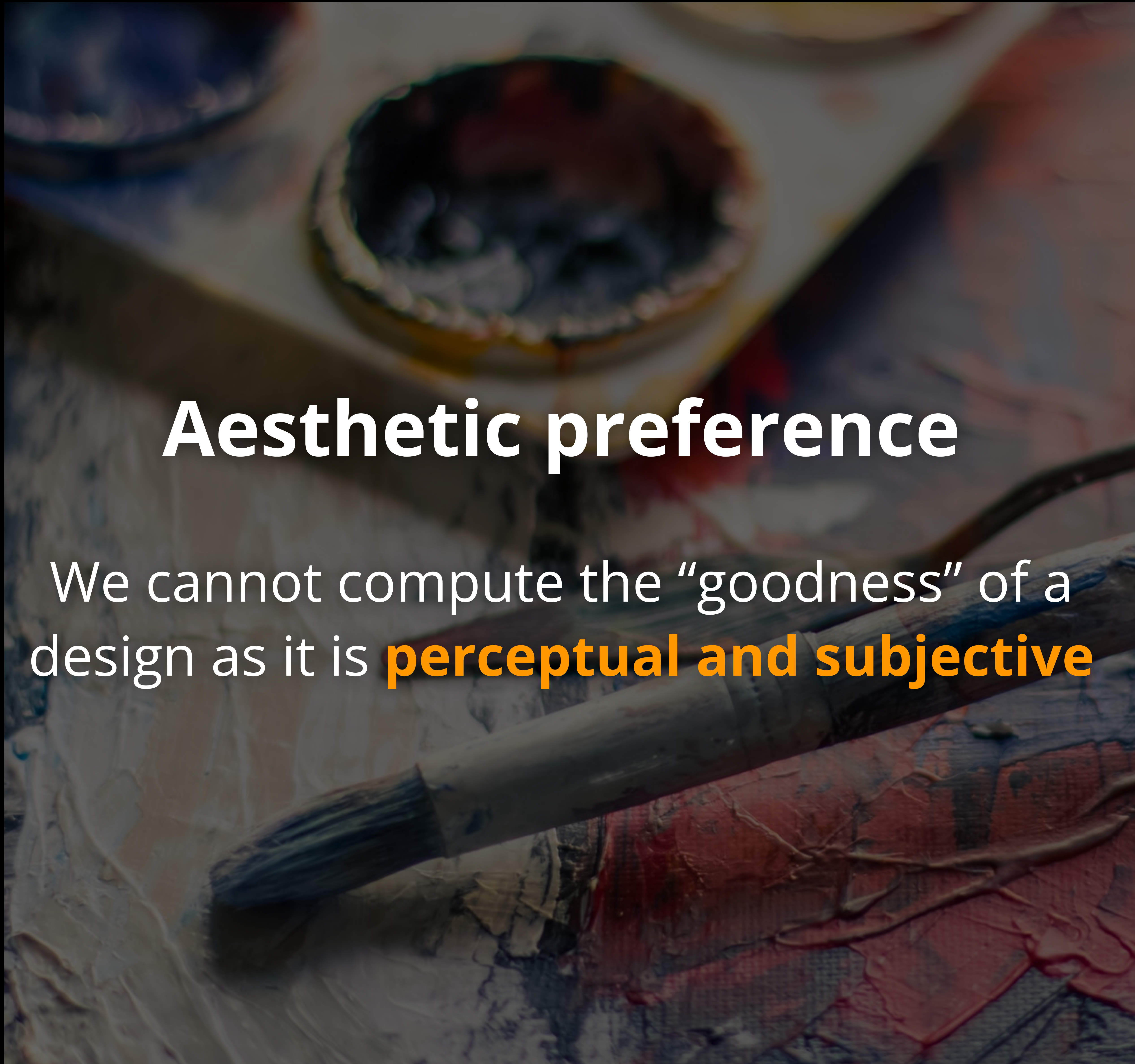
Design Goals (Objectives)



Design Goals (Objectives)

Functionality

We can compute the “goodness” of a design by predictive simulation, etc.



Aesthetic preference

We cannot compute the “goodness” of a design as it is **perceptual and subjective**

Design Goals (Objectives)

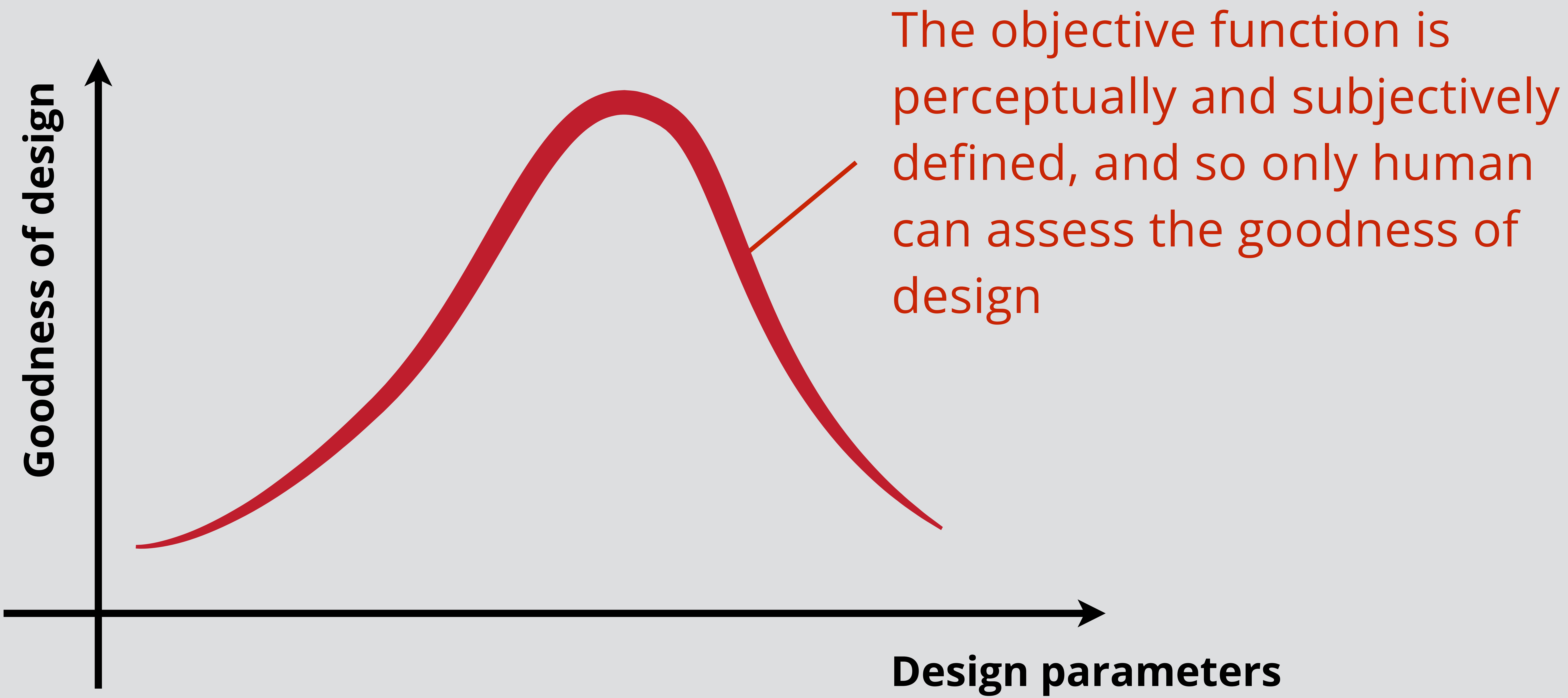
Functionality

We can compute the “goodness” of a design by predictive simulation, etc.

Aesthetic preference

We cannot compute the “goodness” of a design as it is **perceptual and subjective**

Human assessment is necessary


Goodness of design
Design parameters
The objective function is perceptually and subjectively defined, and so only human can assess the goodness of design

Systematic Human Assessment

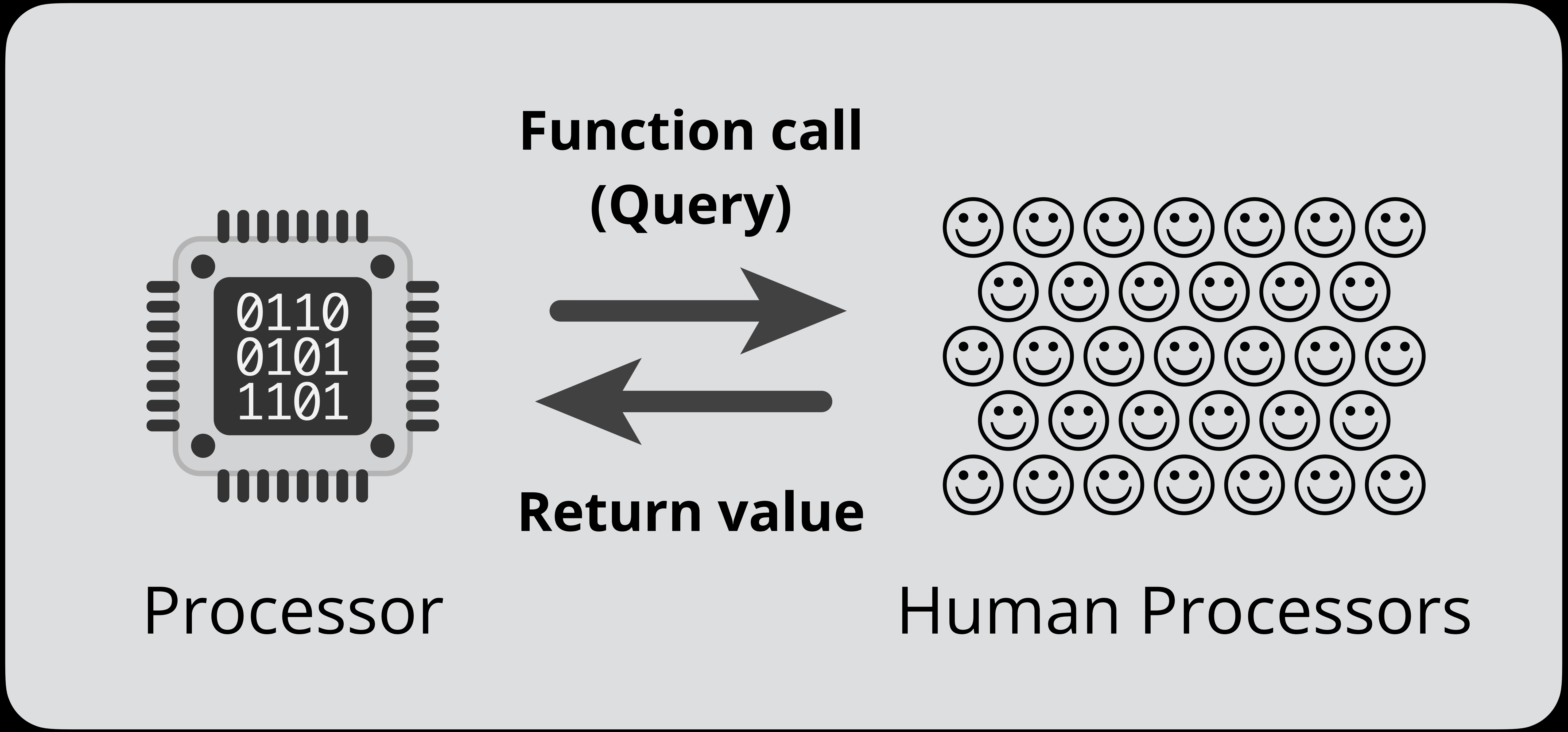
A Systematic Approach to Input Human Assessment

A. J. Quinn and B. B. Bederson. 2011. Human computation: a survey and taxonomy of a growing field. In Proc. CHI '11. 1403-1412.
von Ahn, L. Human Computation. Doctoral Thesis. UMI Order Number: AAI3205378, CMU, (2005).

A Systematic Approach to Input Human Assessment

Human Computation:

"[...] *a paradigm for utilizing human processing power to solve problems that computers cannot yet solve*." [von Ahn 05]

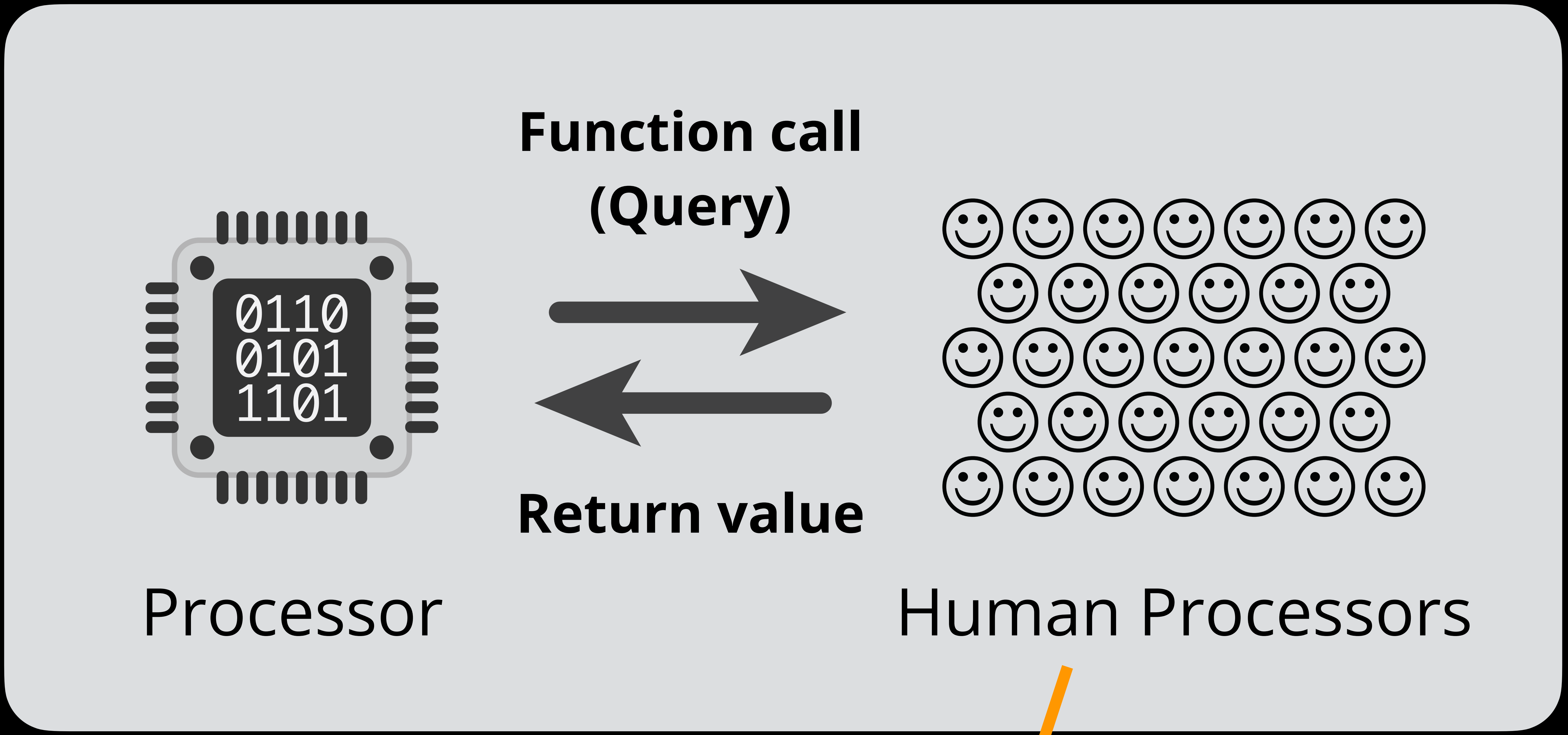


A. J. Quinn and B. B. Bederson. 2011. Human computation: a survey and taxonomy of a growing field. In Proc. CHI '11. 1403-1412.

von Ahn, L. Human Computation. Doctoral Thesis. UMI Order Number: AAI3205378, CMU, (2005).

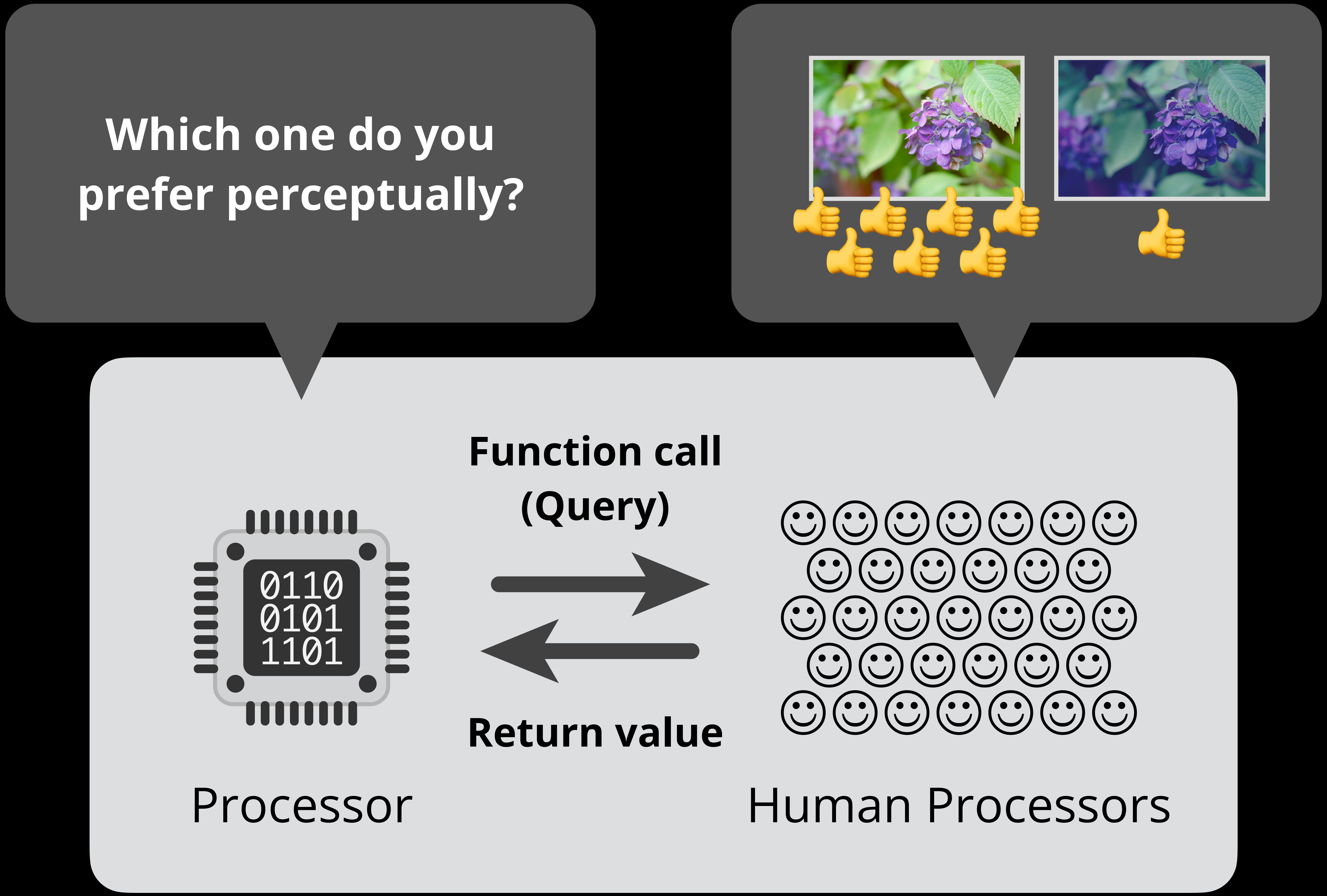
A Systematic Approach to Input Human Assessment

Human Computation:
"[...] *a paradigm for utilizing human processing power to solve problems that computers cannot yet solve*." [von Ahn 05]

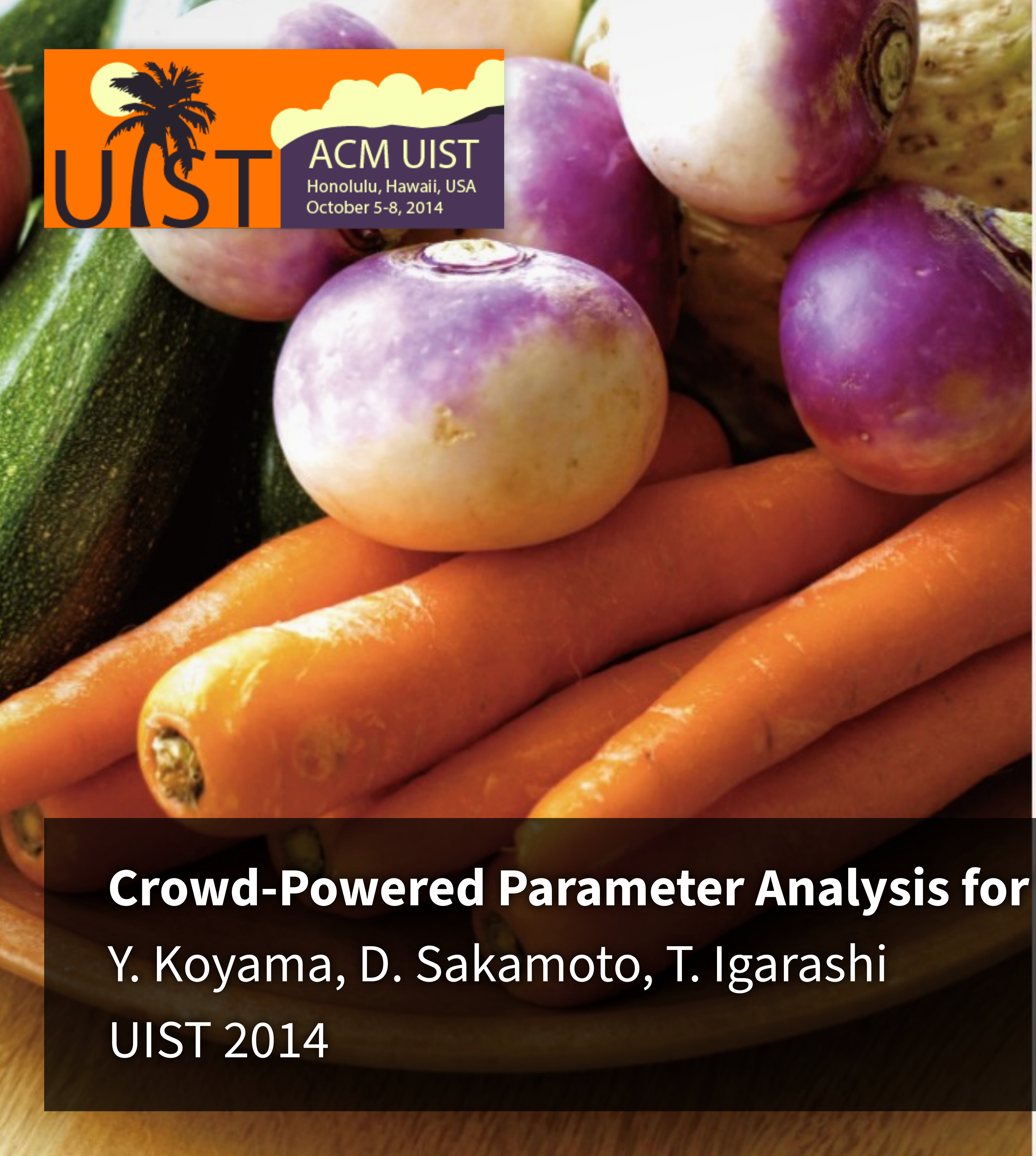


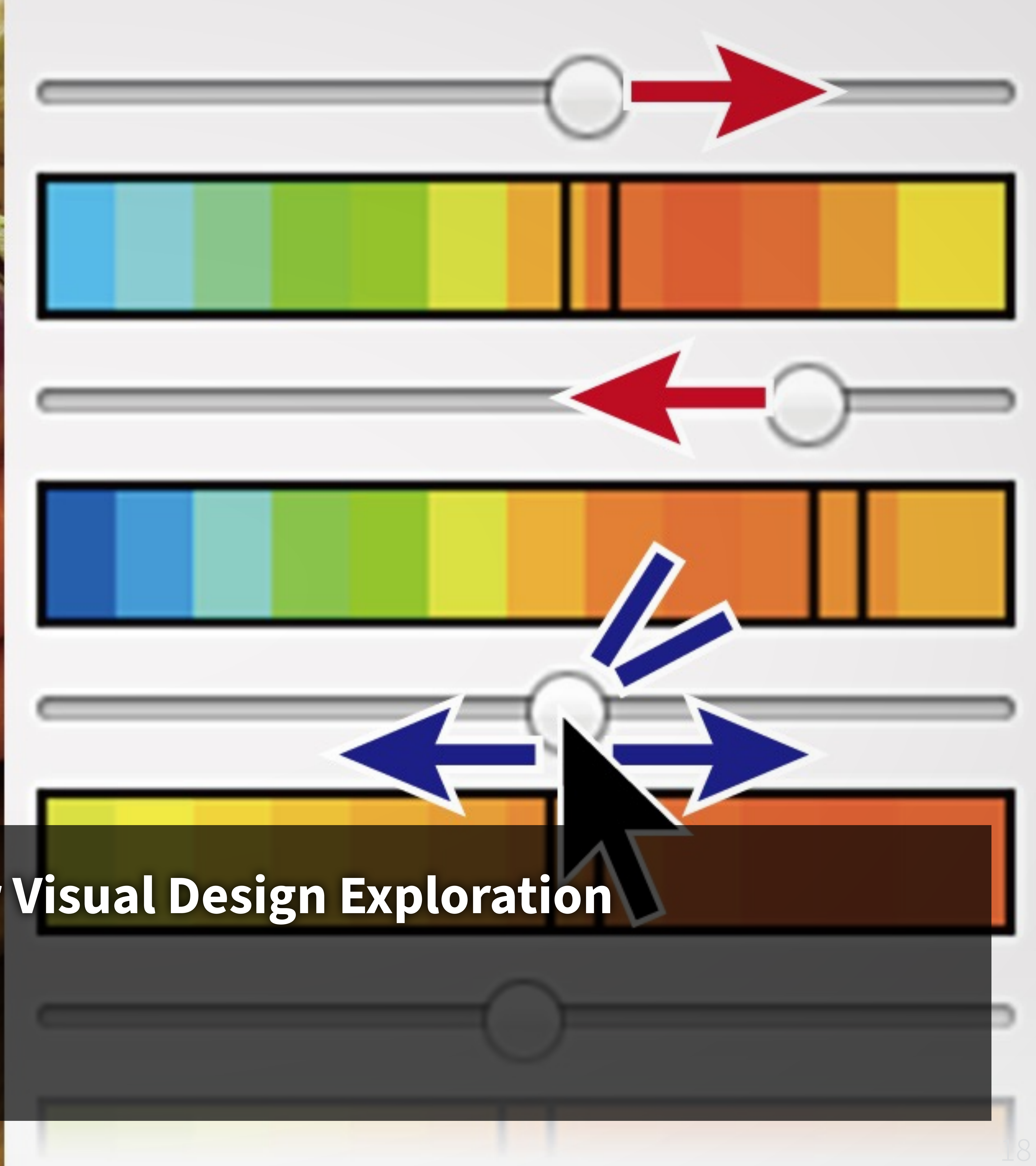
Either crowd workers or the single user

A. J. Quinn and B. B. Bederson. 2011. Human computation: a survey and taxonomy of a growing field. In Proc. CHI '11. 1403-1412.
von Ahn, L. Human Computation. Doctoral Thesis. UMI Order Number: AAI3205378, CMU, (2005).



A. J. Quinn and B. B. Bederson. 2011. Human computation: a survey and taxonomy of a growing field. In Proc. CHI '11. 1403-1412.
von Ahn, L. Human Computation. Doctoral Thesis. UMI Order Number: AAI3205378, CMU, (2005).





ACM UIST
Honolulu, Hawaii, USA
October 5-8, 2014

Crowd-Powered Parameter Analysis for Visual Design Exploration

Y. Koyama, D. Sakamoto, T. Igarashi

UIST 2014

Parametric design is ubiquitous in digital content creation

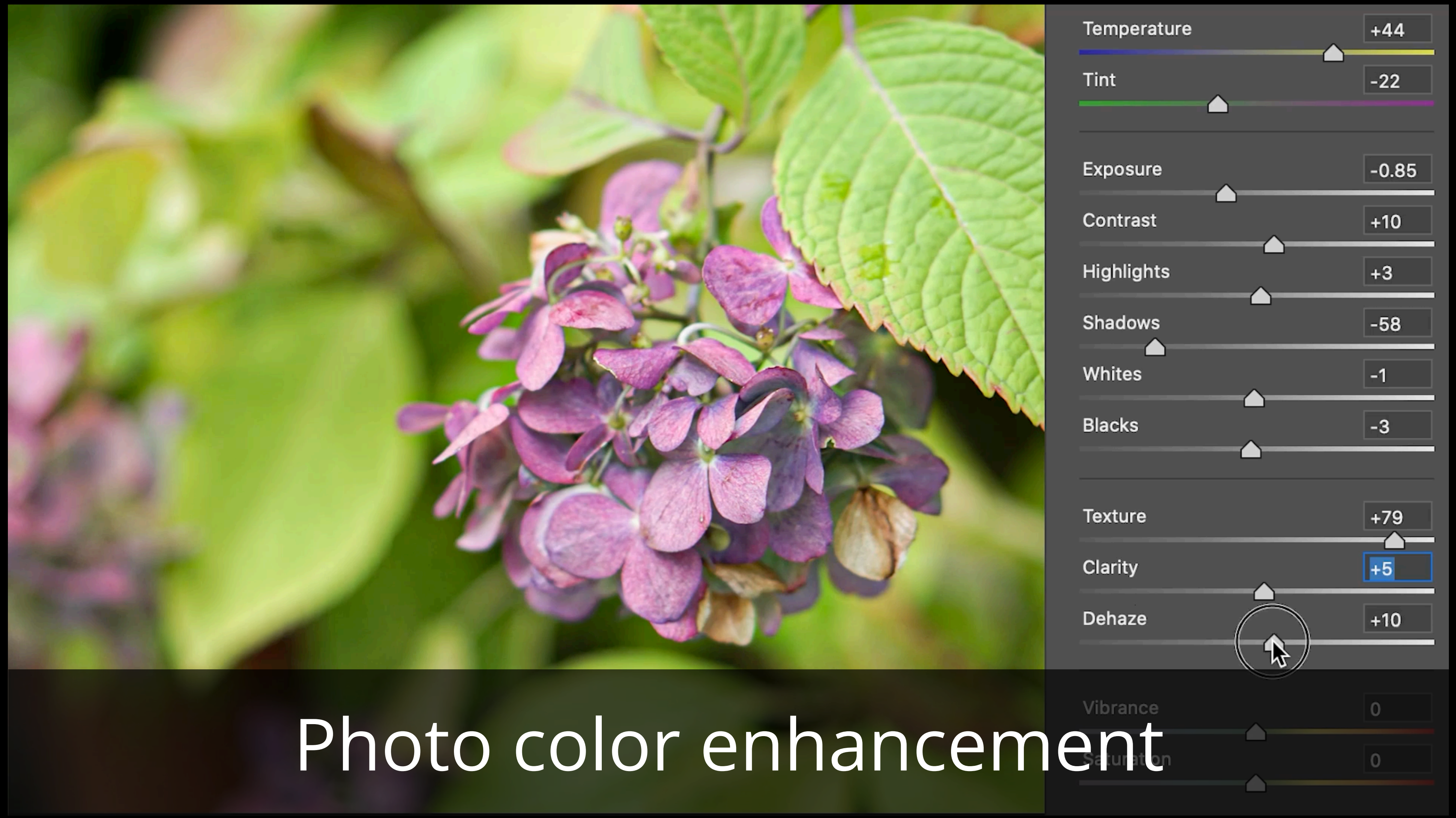
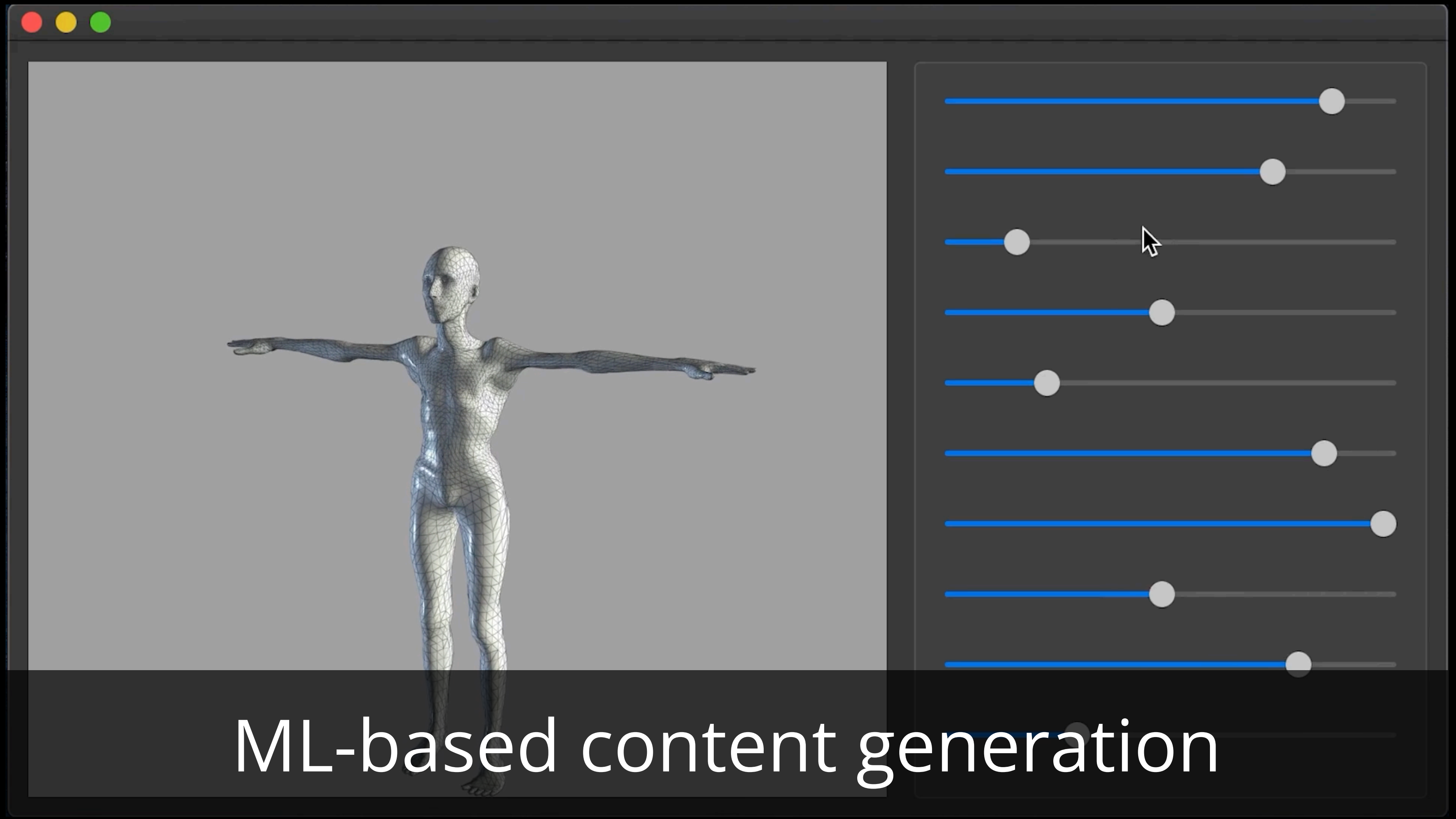
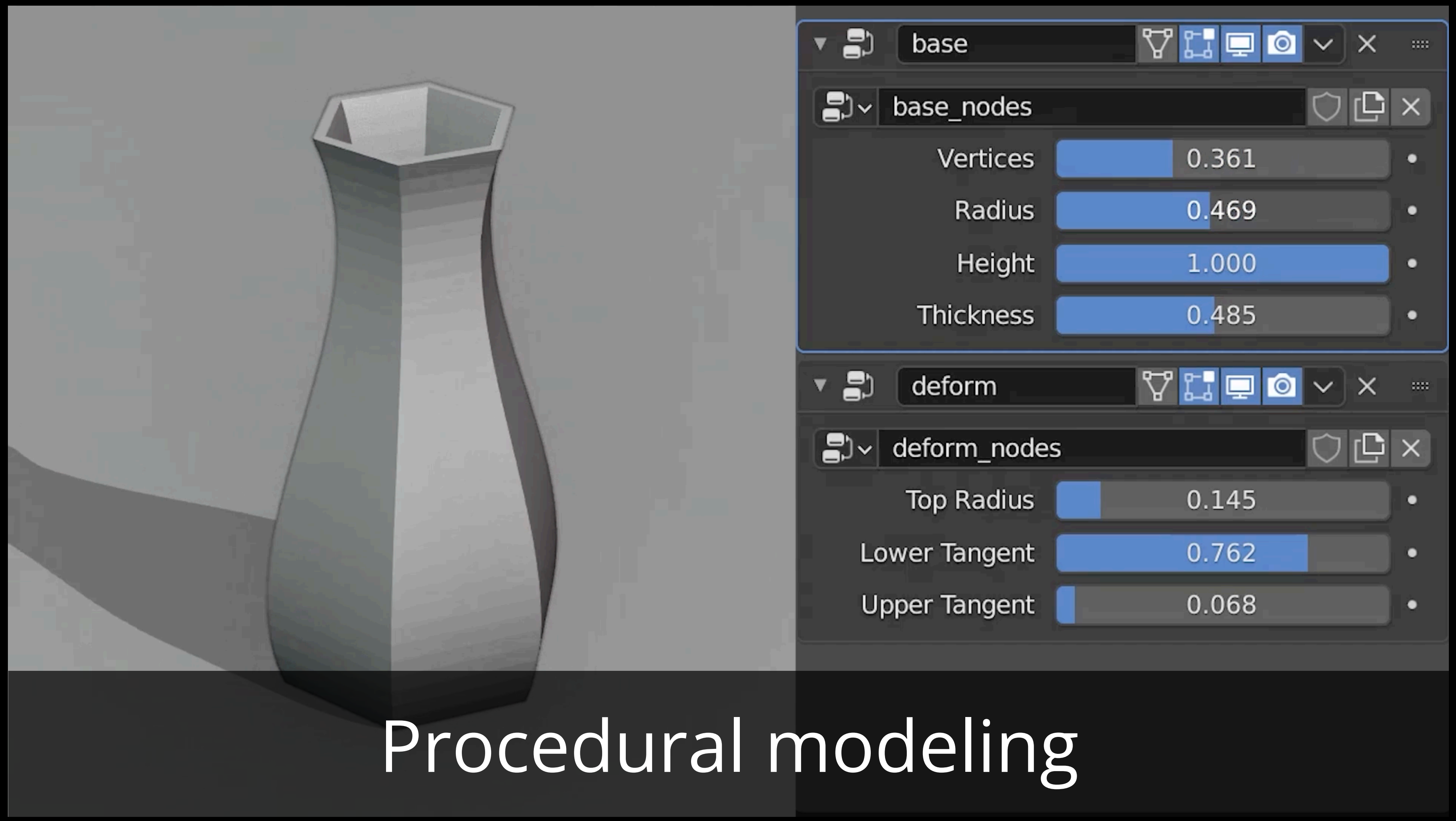
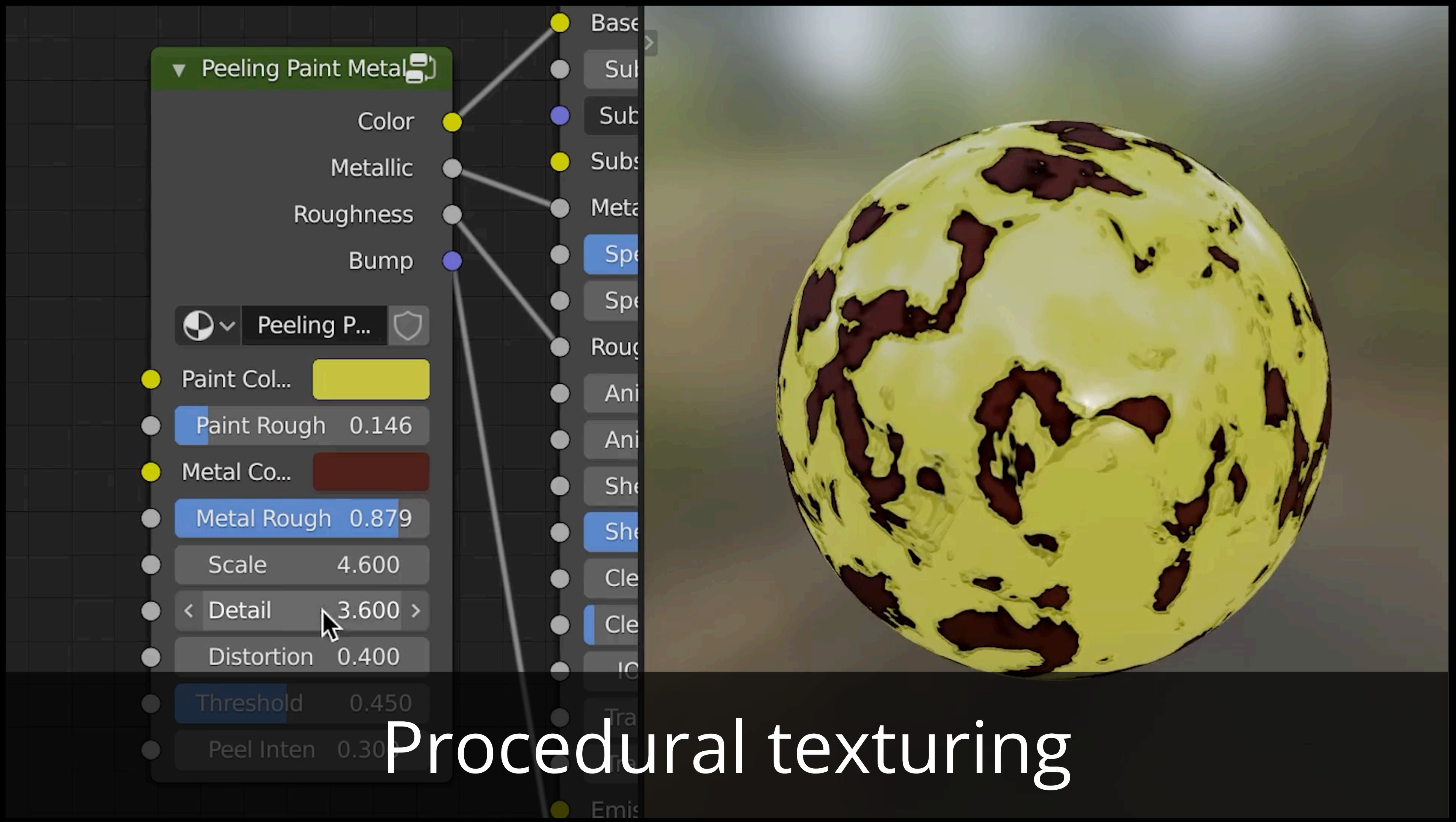


Photo color enhancement


ML-based content generation

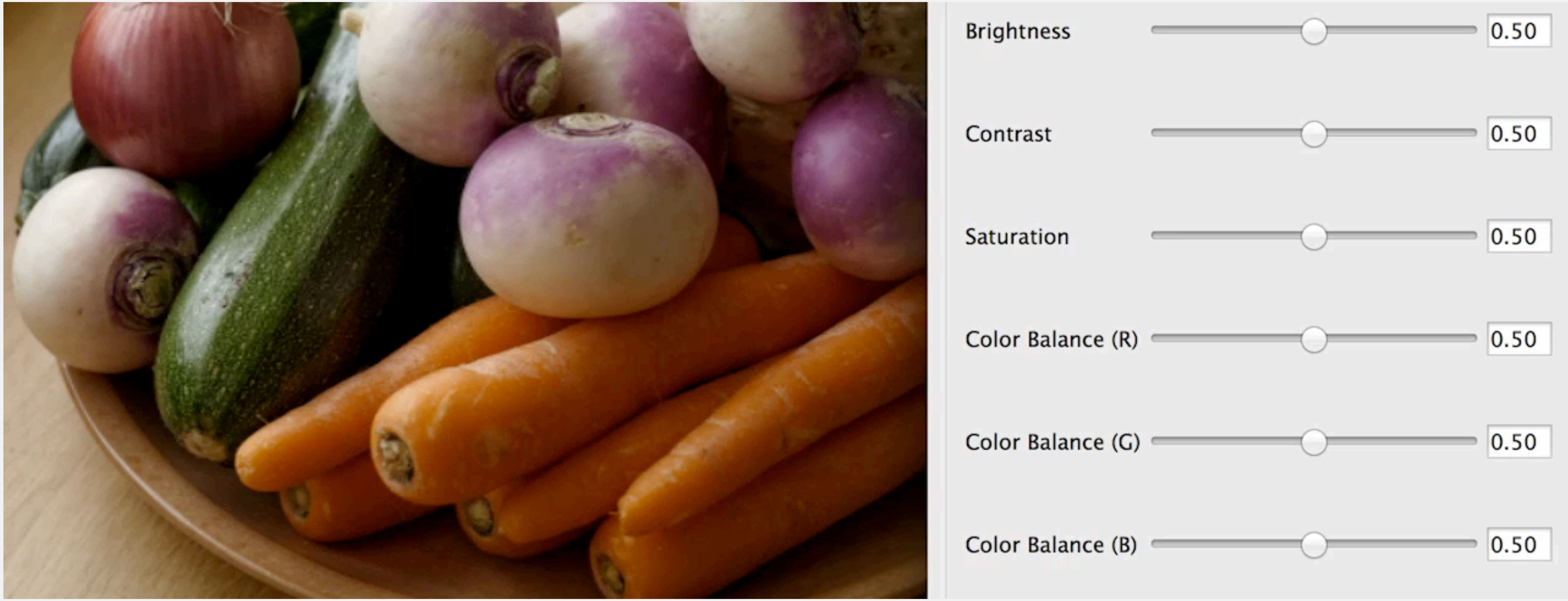


Procedural modeling

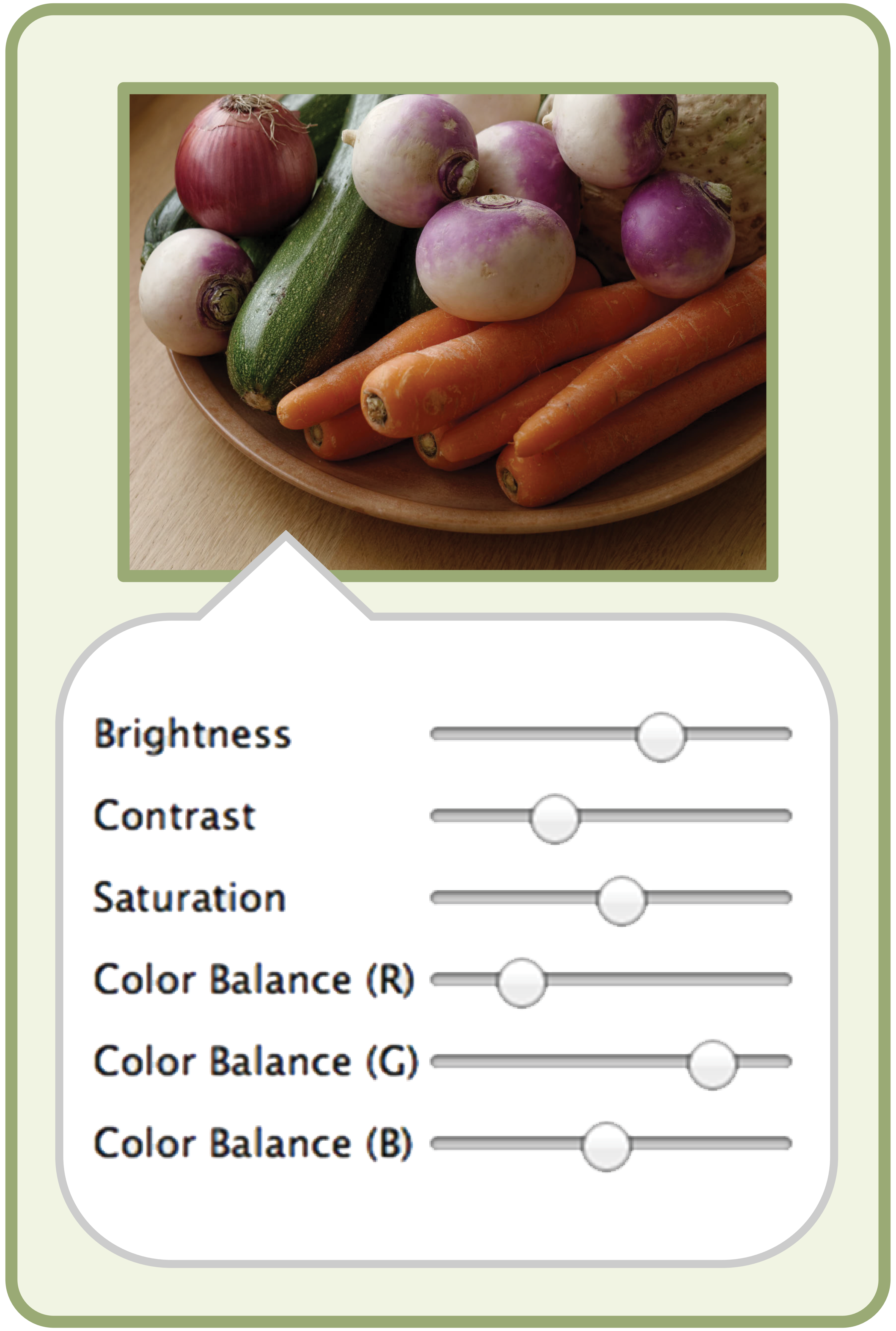


Procedural texturing

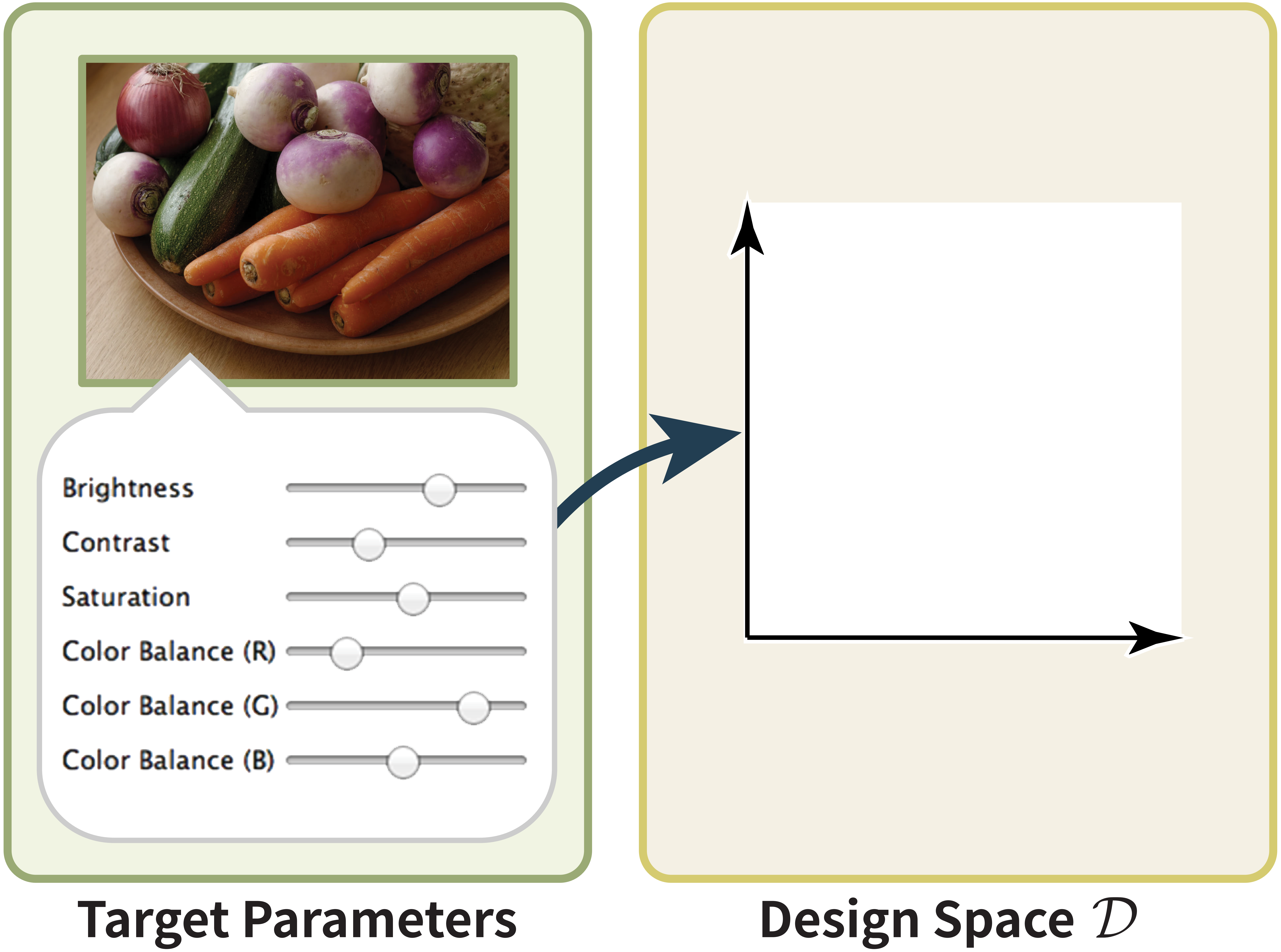
Goal: Search for the Best Preferable Design Parameters

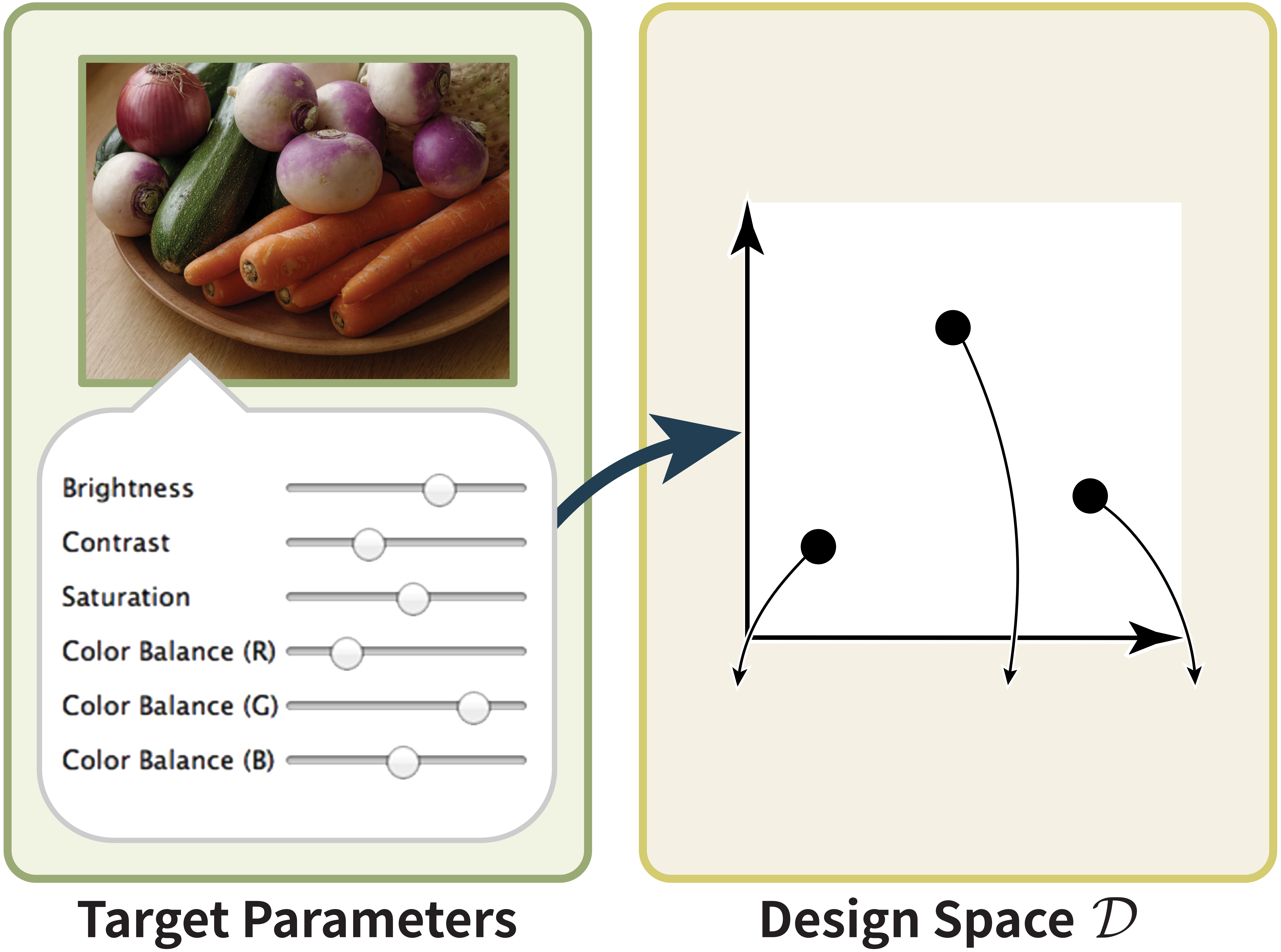


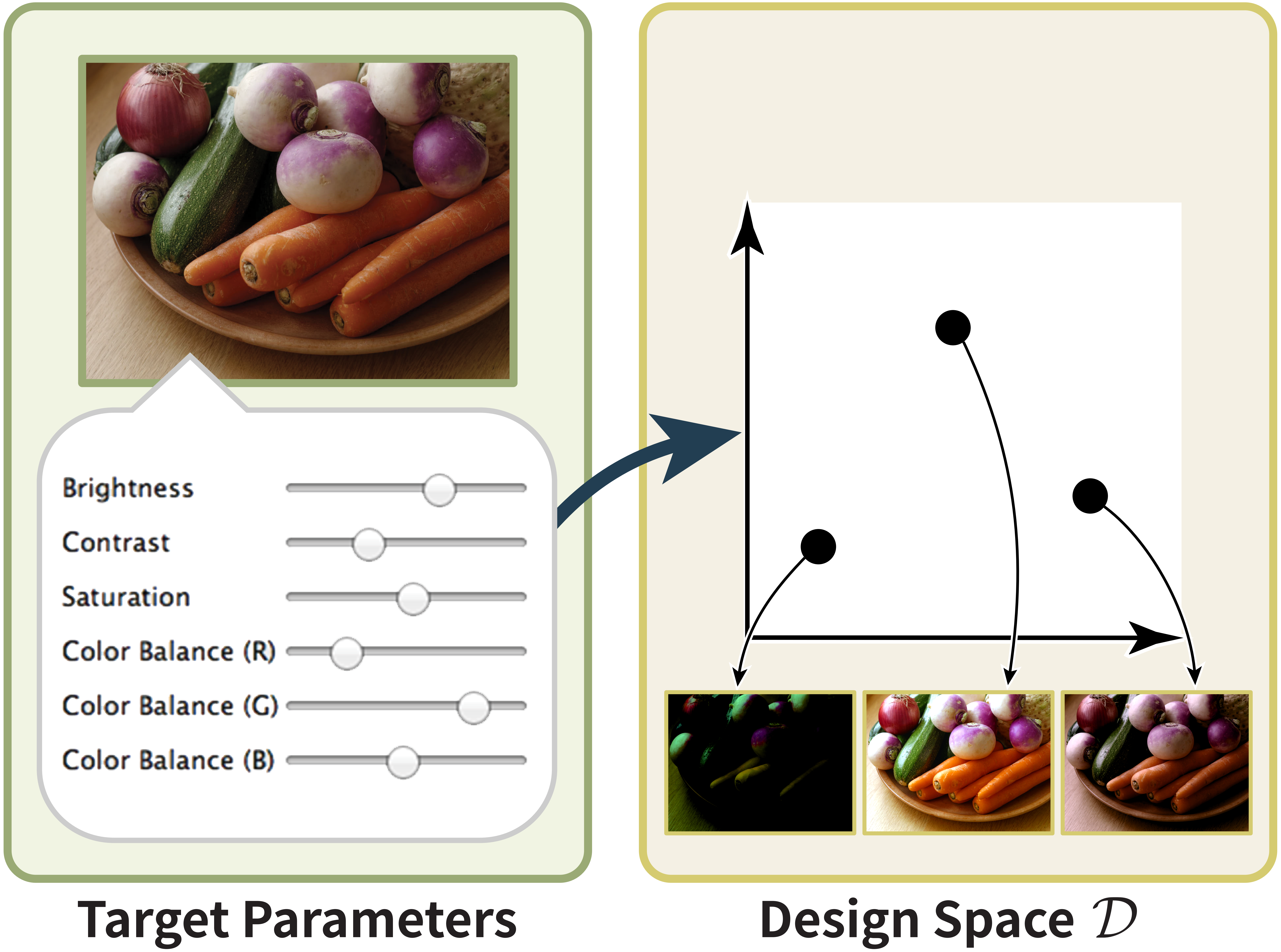
E.g., photo color enhancement

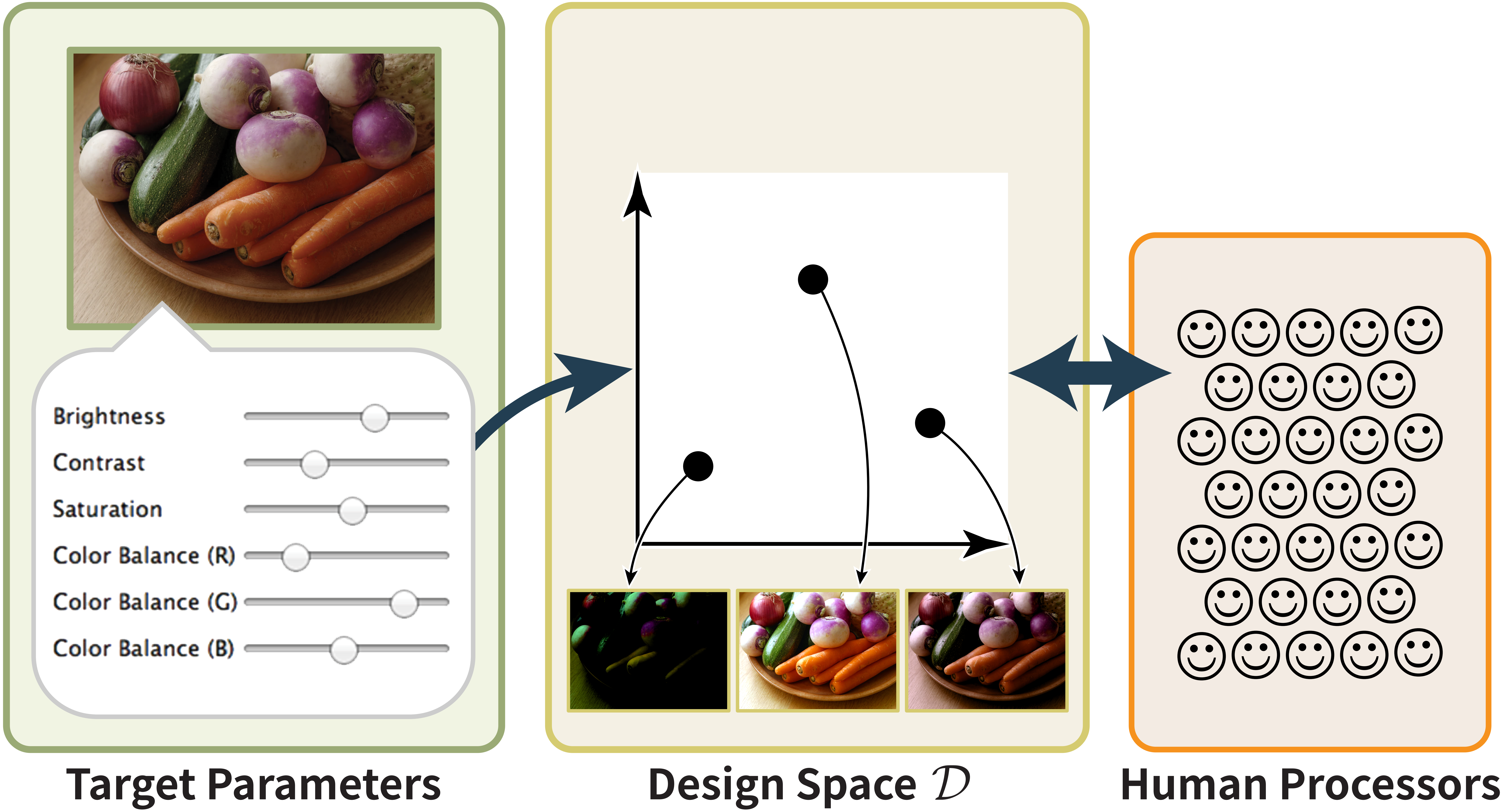


Target Parameters


Brightness
Contrast
Saturation
Color Balance (R)
Color Balance (G)
Color Balance (B)
Target Parameters
Design Space $\mathcal{D}$


Brightness
Contrast
Saturation
Color Balance (R)
Color Balance (G)
Color Balance (B)
Target Parameters
Design Space $\mathcal{D}$


Brightness
Contrast
Saturation
Color Balance (R)
Color Balance (G)
Color Balance (B)
Target Parameters
Design Space $\mathcal{D}$



Brightness
Contrast
Saturation
Color Balance (R)
Color Balance (G)
Color Balance (B)
Target Parameters
Design Space $\mathcal{D}$
Human Processors

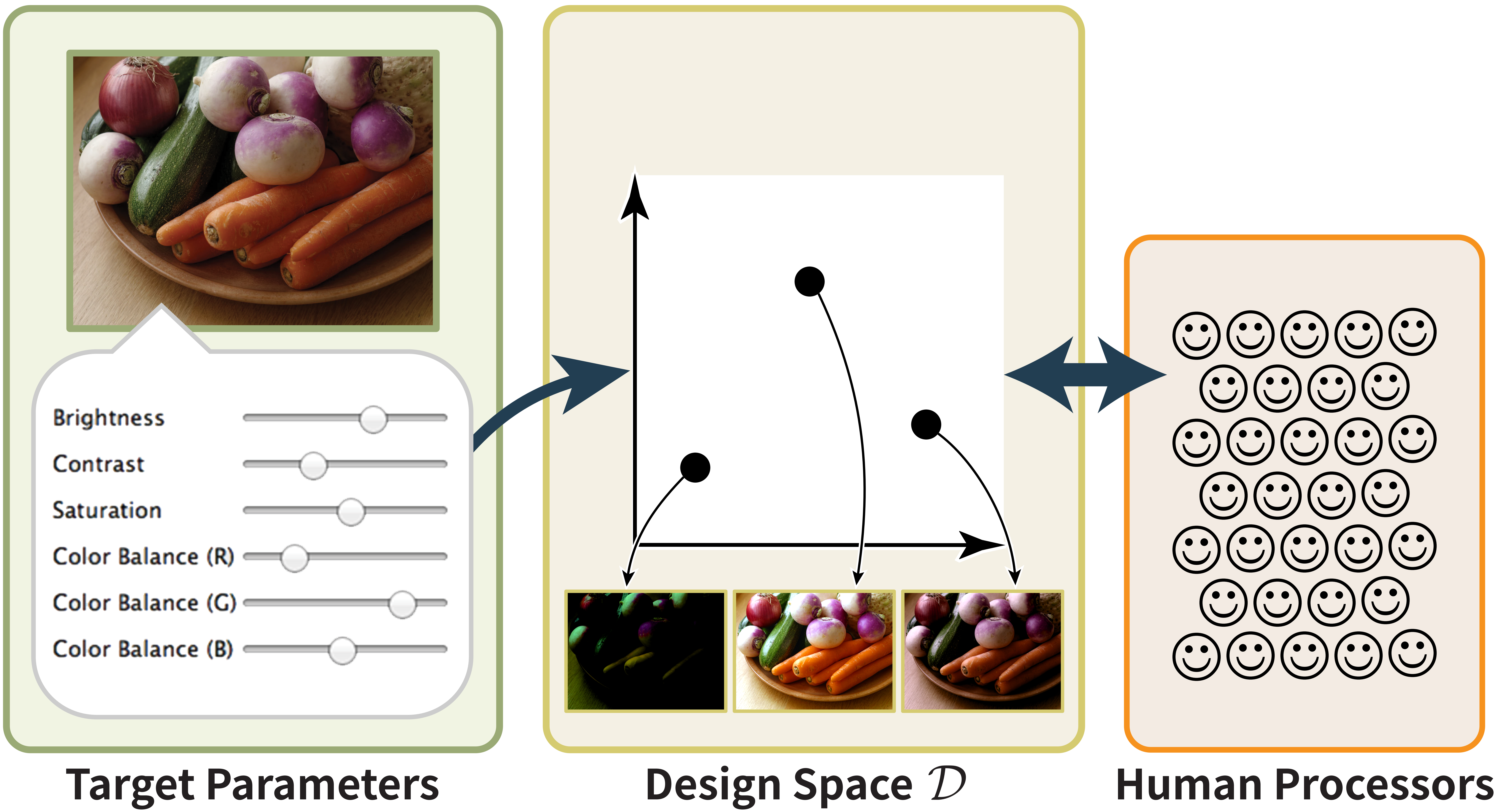
Preference data generation [pairwise comparison]

(c.f., 2000 comparisons, 4 USD, 30 min)

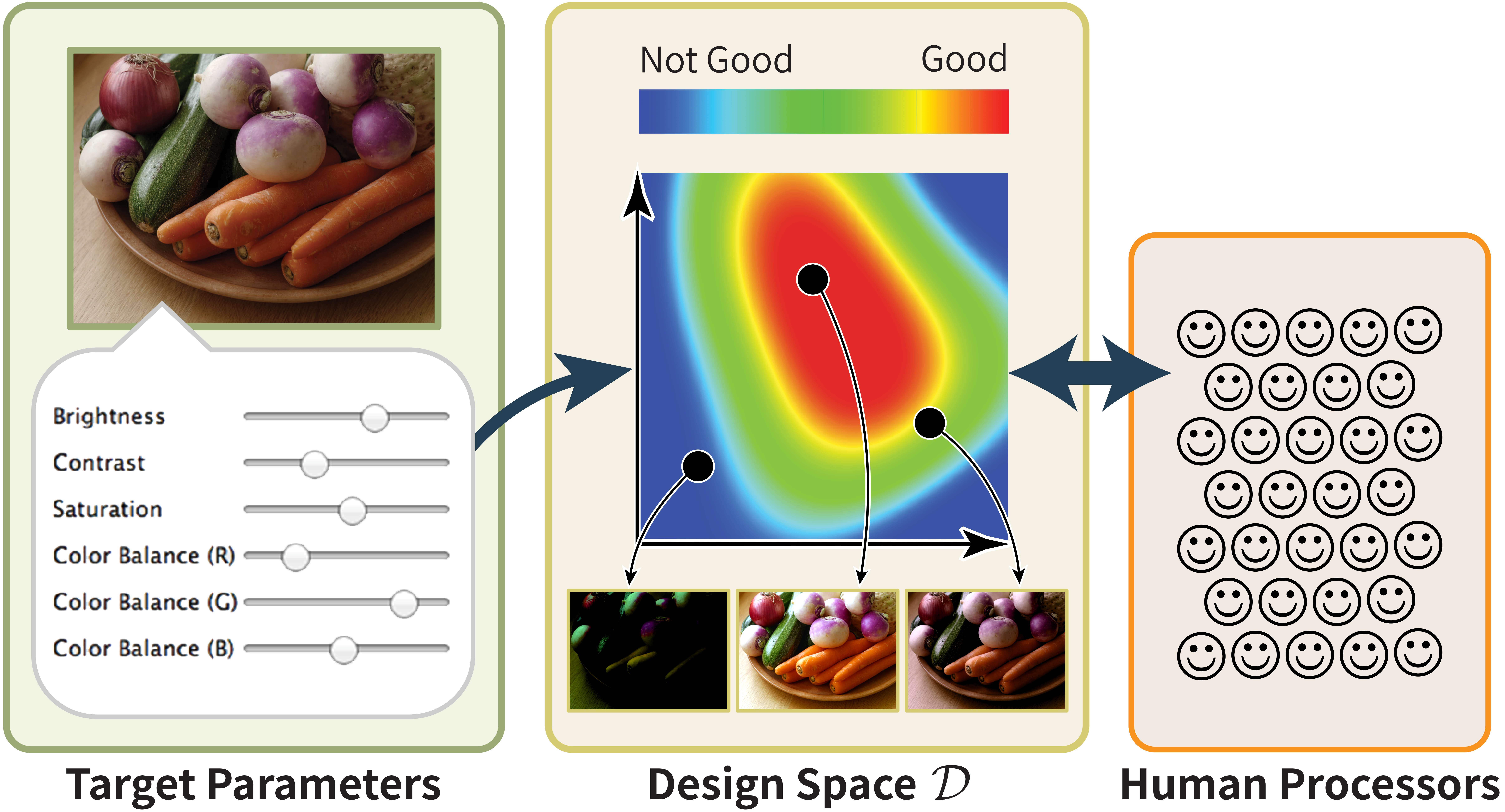
Target Parameters

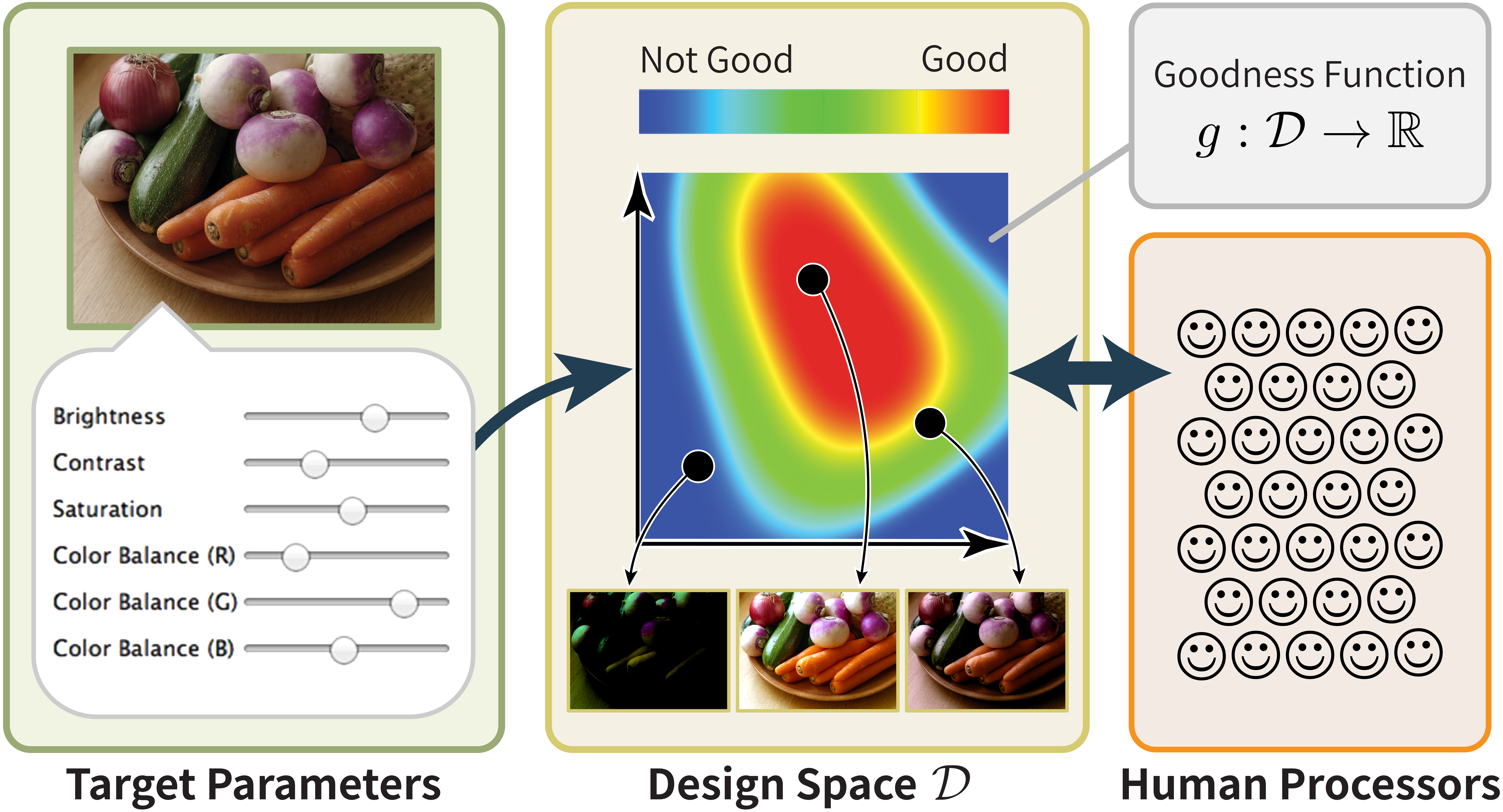
Design Space

Human Processors

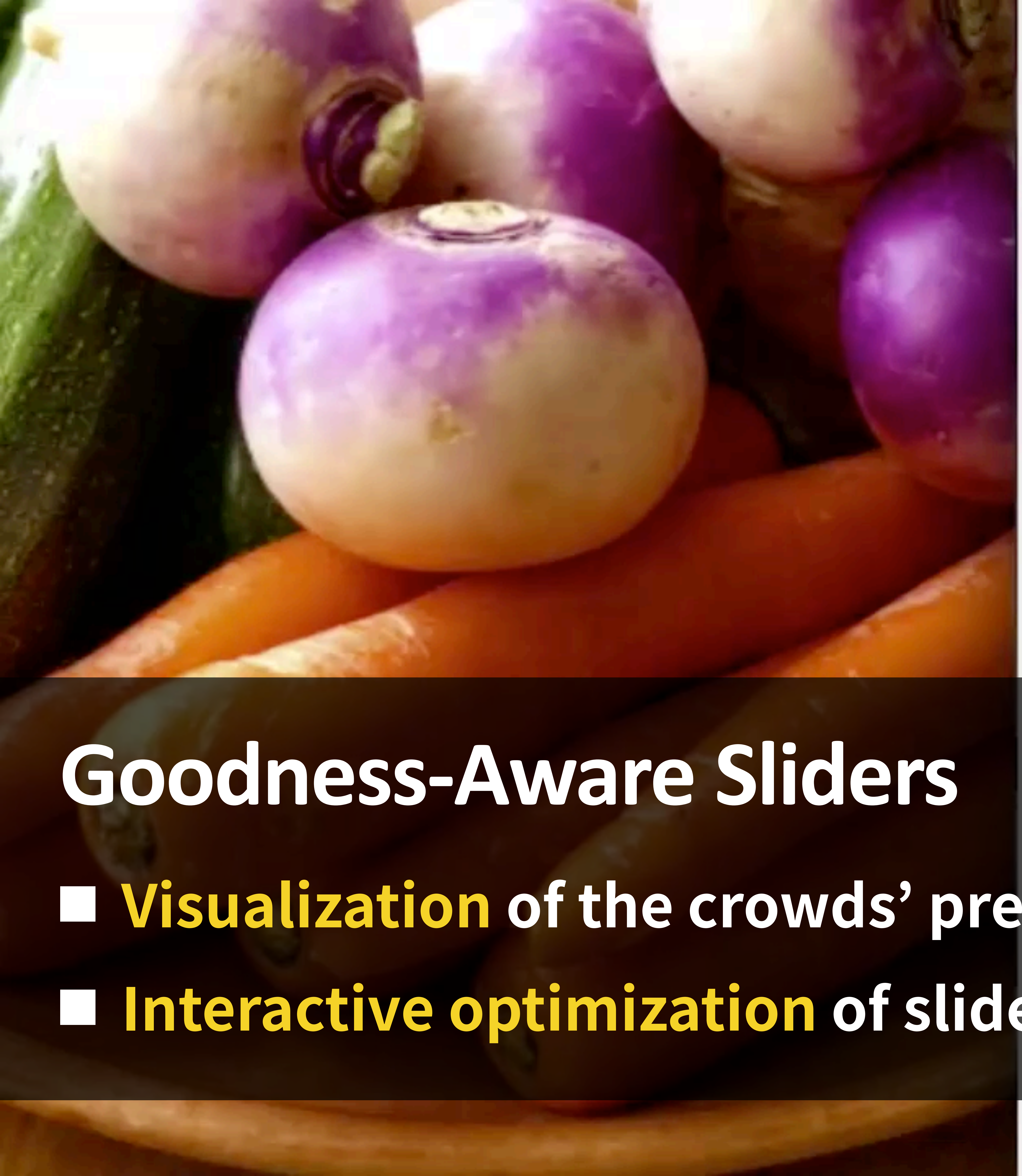


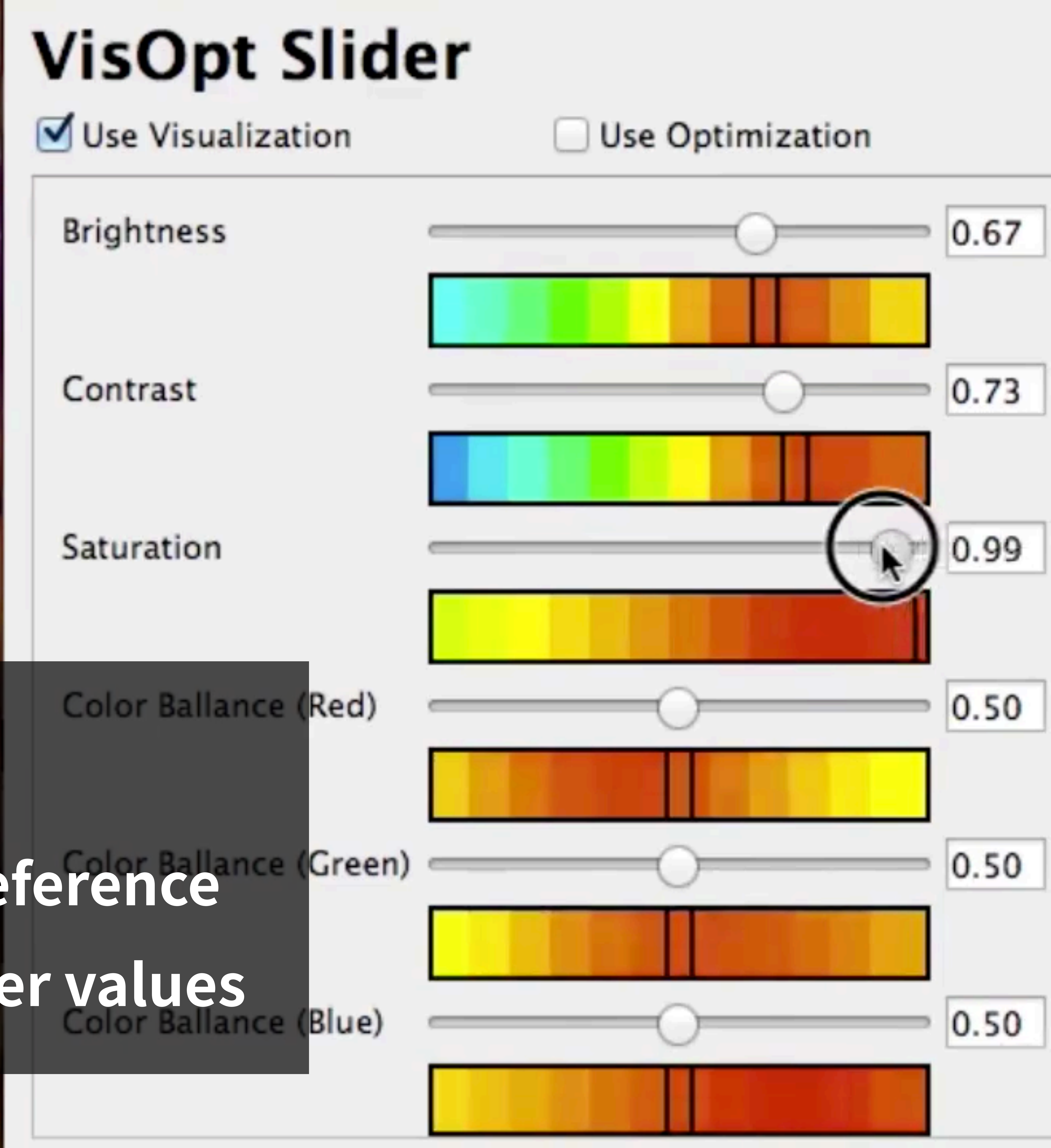
Brightness
Contrast
Saturation
Color Balance (R)
Color Balance (G)
Color Balance (B)
Target Parameters
Design Space $\mathcal{D}$
Human Processors


Brightness
Contrast
Saturation
Color Balance (R)
Color Balance (G)
Color Balance (B)
Not Good
Good
Target Parameters
Design Space $\mathcal{D}$
Human Processors



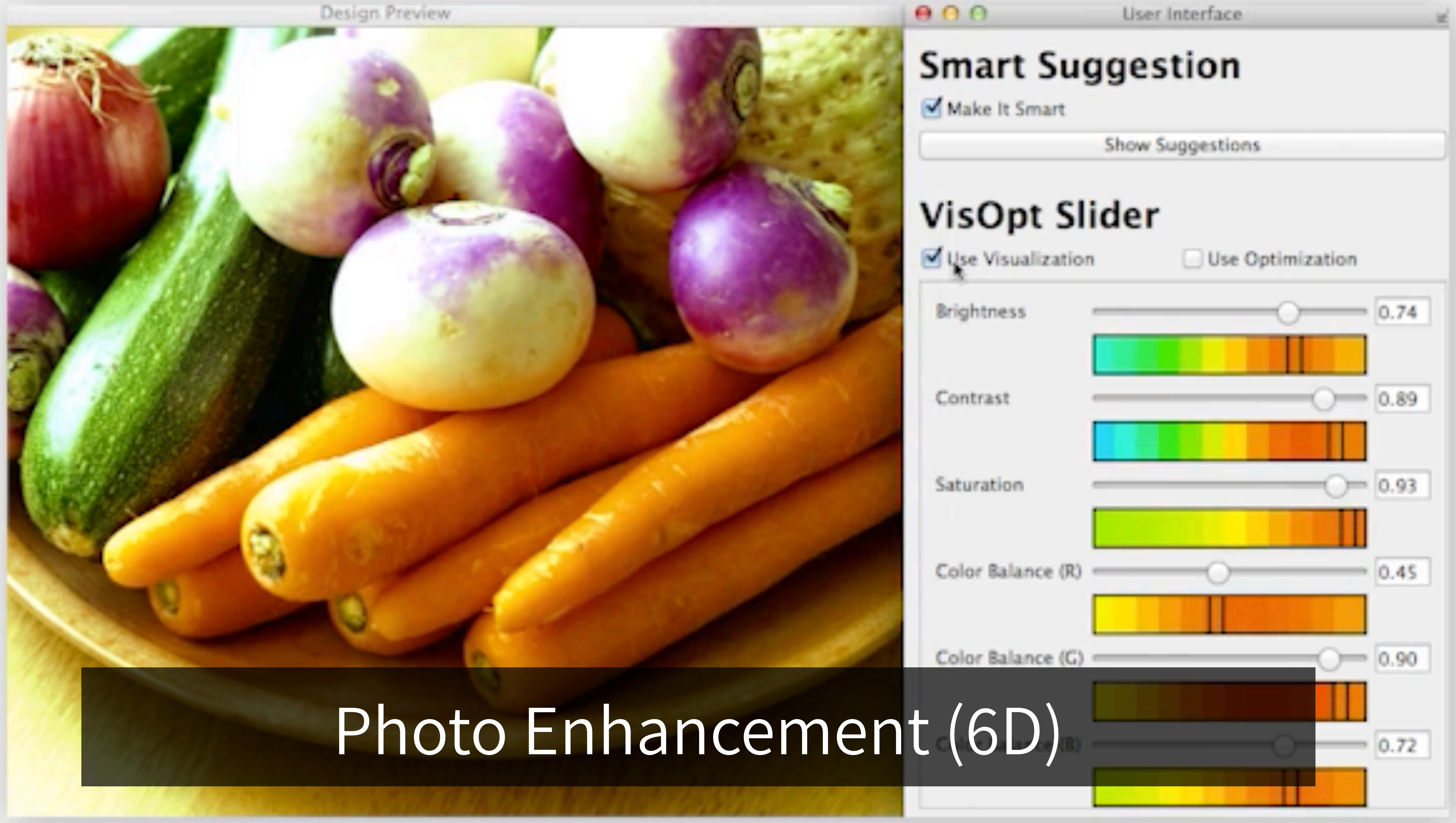
Not Good
Good
Goodness Function
$g : \mathcal{D} \to \mathbb{R}$
Brightness
Contrast
Saturation
Color Balance (R)
Color Balance (G)
Color Balance (B)
Target Parameters
Design Space $\mathcal{D}$
Human Processors

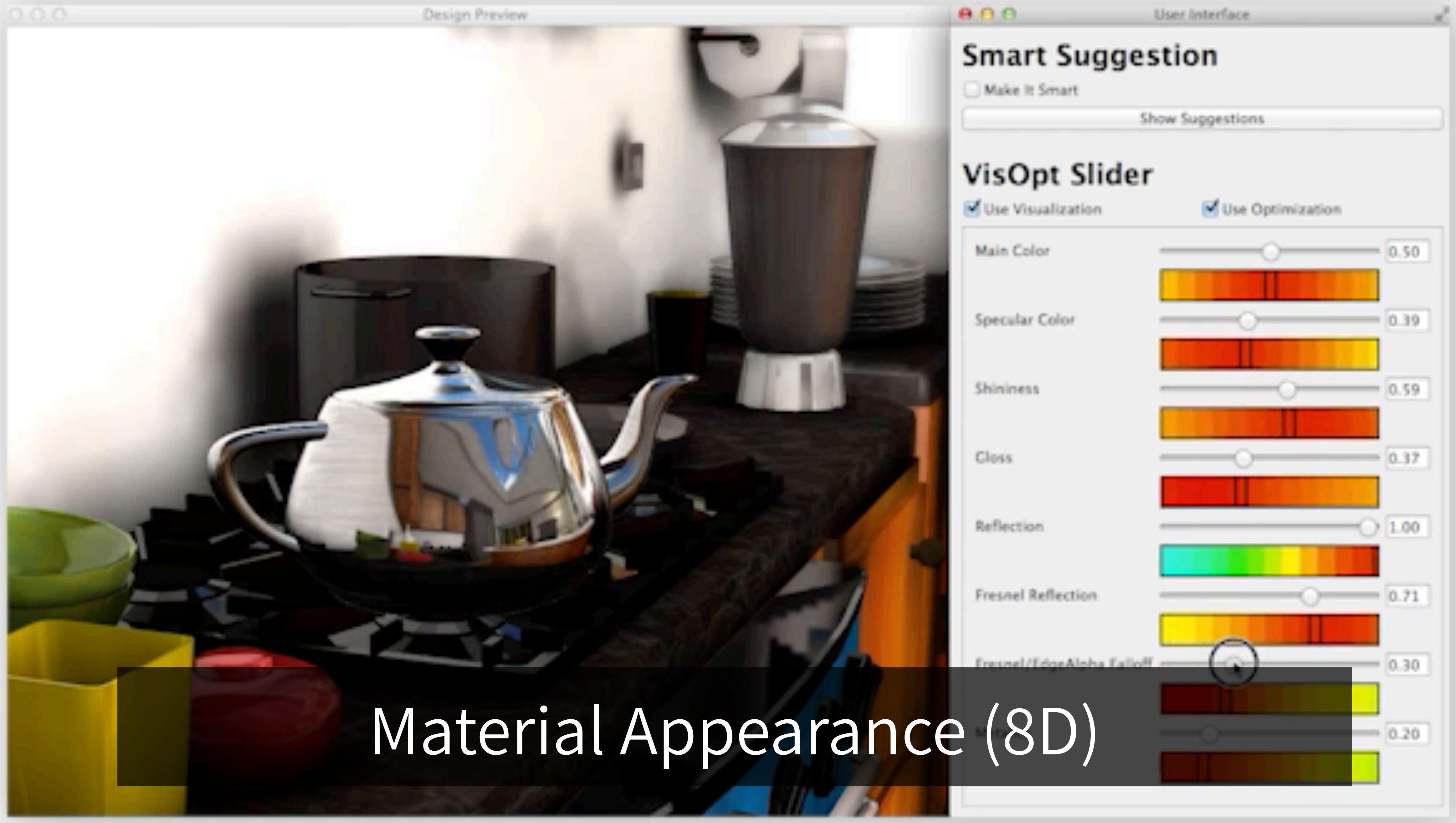


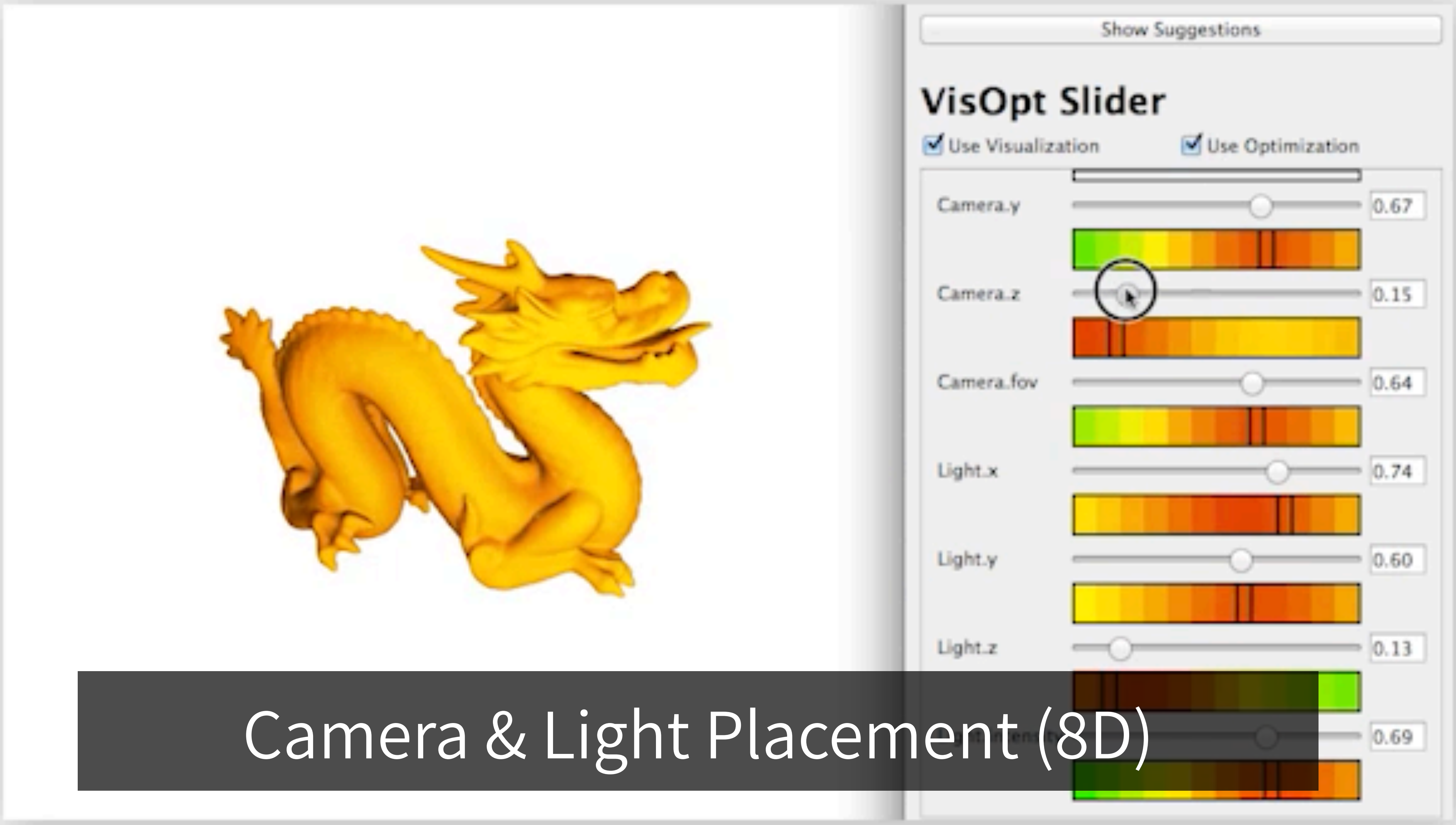


Goodness-Aware Sliders

- **Visualization** of the crowds' preference
- **Interactive optimization** of slider values


Design Preview
User Interface
Smart Suggestion
Make It Smart
Show Suggestions
VisOpt Slider
Use Visualization
Use Optimization
Brightness
0.74
Contrast
0.89
Saturation
0.93
Color Balance (R)
0.45
Color Balance (G)
0.90
0.72
Photo Enhancement (6D)


Design Preview
User Interface
Smart Suggestion
Make It Smart
Show Suggestions
VisOpt Slider
Use Visualization
Use Optimization
Main Color
0.50
Specular Color
0.39
Shininess
0.59
0.37
Reflection
1.00
Fresnel Reflection
0.71
0.30
0.20
Material Appearance (8D)


Show Suggestions
VisOpt Slider
Use Visualization
Use Optimization
Camera.y
0.67
Camera.z
0.15
Camera.fov
0.64
Light.x
0.74
Light.y
0.60
Light.z
0.13
0.69
Camera & Light Placement (8D)


Design Preview
User Interface
Smart Suggestion
Make It Smart
Show Suggestions
VisOpt Slider
Use Visualization
Use Optimization
BrowsD_L
0.49
BrowsD_R
0.43
BrowsU_C
0.04
BrowsU_L
0.92
BrowsU_R
0.95
CheekSquint_L
0.35
0.45
0.26
Facial Expression (53D)

$$\max_{\mathbf{x}} \ \mathrm{Preference}(\mathbf{x})$$

$$\max_{\mathbf{x}} \ \text{Preference}(\mathbf{x})$$

Computationally estimate the perceptual function shape

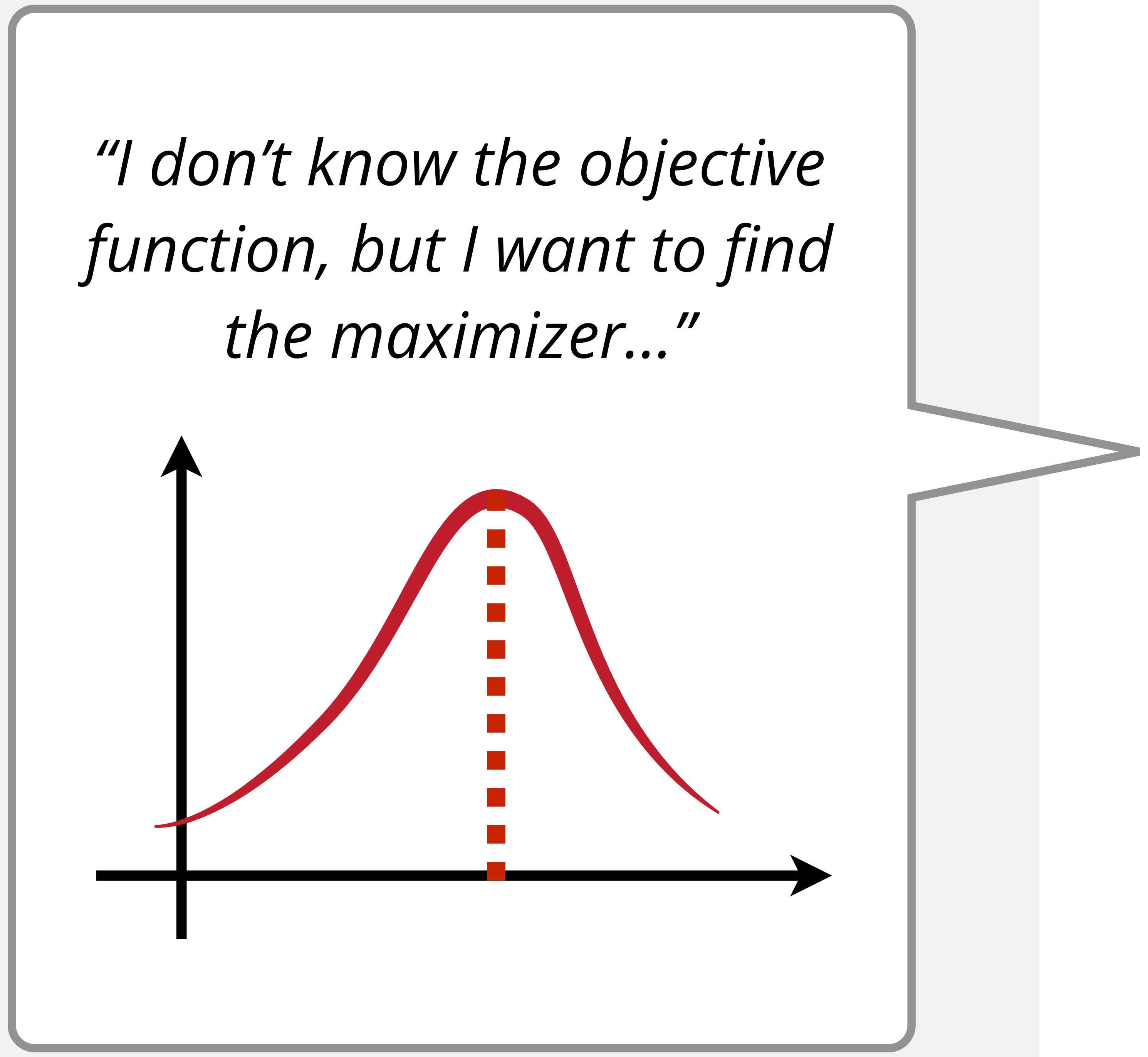
➡ Support **the user's manual search**

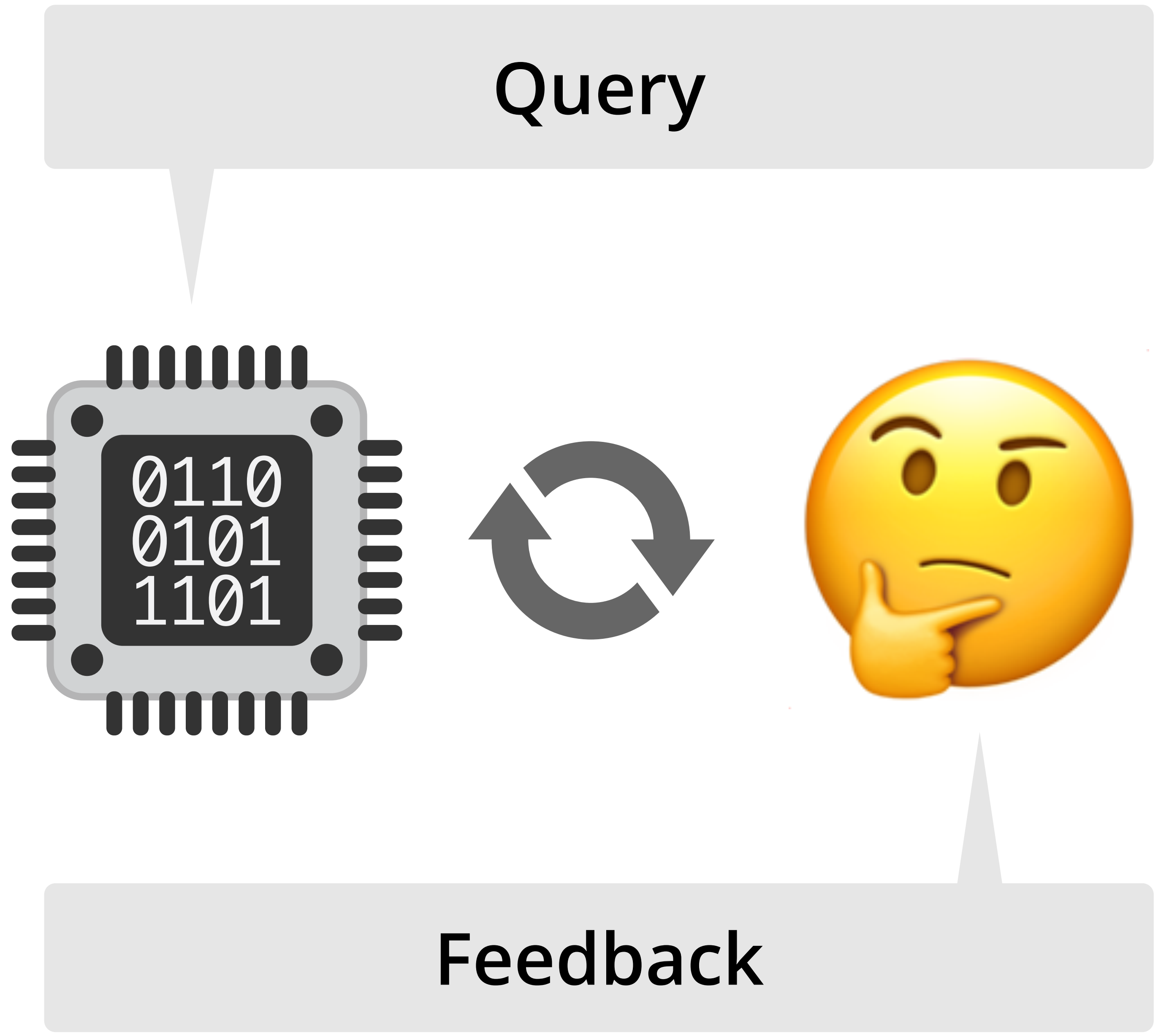
$$\max_{\mathbf{x}} \ \mathrm{Preference}(\mathbf{x})$$

Is it possible to computationally execute the search itself?

Human-in-the-Loop Optimization

Human-in-the-Loop Optimization





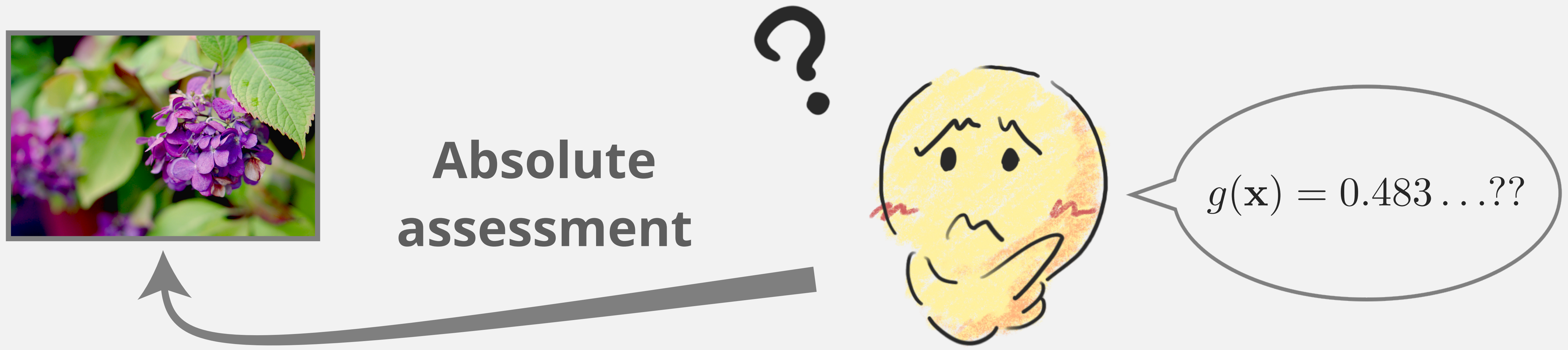
Iterative search process

Human-in-the-loop optimization is used for solving problems with **"subjective" objective functions** (e.g., preference)

Considerations [1/2]: Query Design

Considerations [1/2]: Query Design

Absolute assessment should be avoided: The user cannot reliably answer the function value [Brochu+10; Koyama+18]



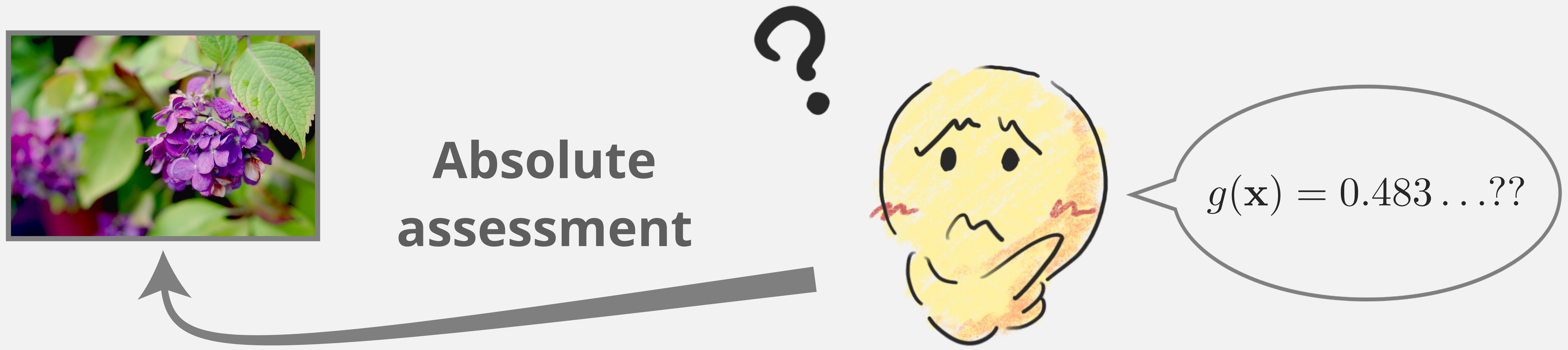
[Koyama+, Computational Interaction (2018)] Yuki Koyama and Takeo Igarashi. 2018. Computational Design with Crowds. In Computational Interaction (Eds. A. Oulasvirta, P. O. Kristensson, X. Bi, and A. Howes), Oxford University Press, pp.153–184. https://arxiv.org/abs/2002.08657

Considerations [1/2]: Query Design

Absolute assessment

should be avoided:

The user cannot reliably answer the function value [Brochu+10; Koyama+18]



Relative assessment

should be used:

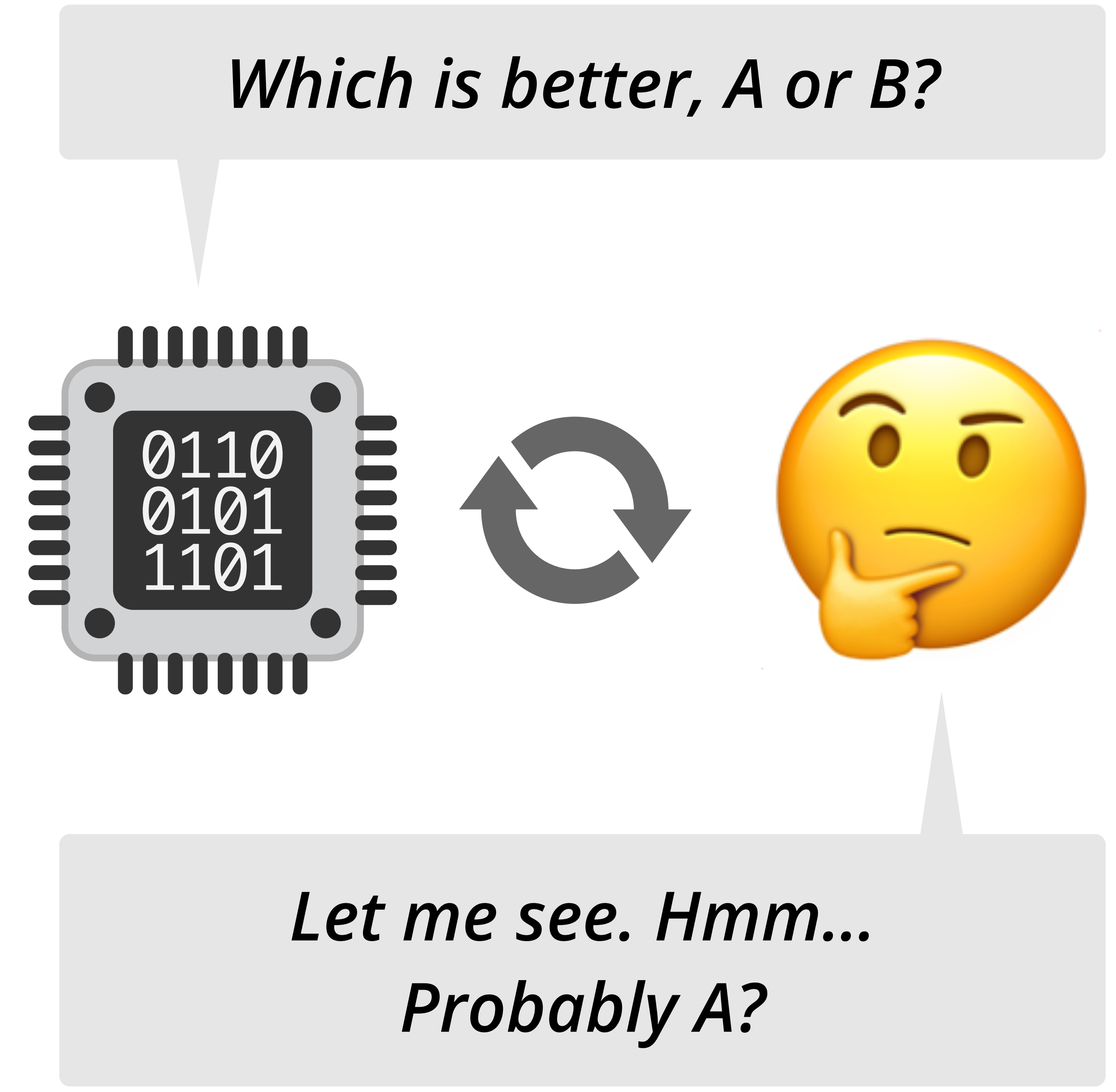
The user can answer which option is better among two (or more) options

[Koyama+, Computational Interaction (2018)] Yuki Koyama and Takeo Igarashi. 2018. Computational Design with Crowds. In Computational Interaction (Eds. A. Oulasvirta, P. O. Kristensson, X. Bi, and A. Howes), Oxford University Press, pp.153–184. https://arxiv.org/abs/2002.08657

Considerations [2/2]: Number of Queries

Considerations [2/2]: Number of Queries

- **The number of total queries should be sufficiently small**
  - Human evaluators will be prohibitively slow (compared to typical optimization settings)
  - Human evaluators will get tired and won't be willing to respond to so many queries



➡ **This takes at least a few seconds**

An Approach: Preferential Bayesian Optimization

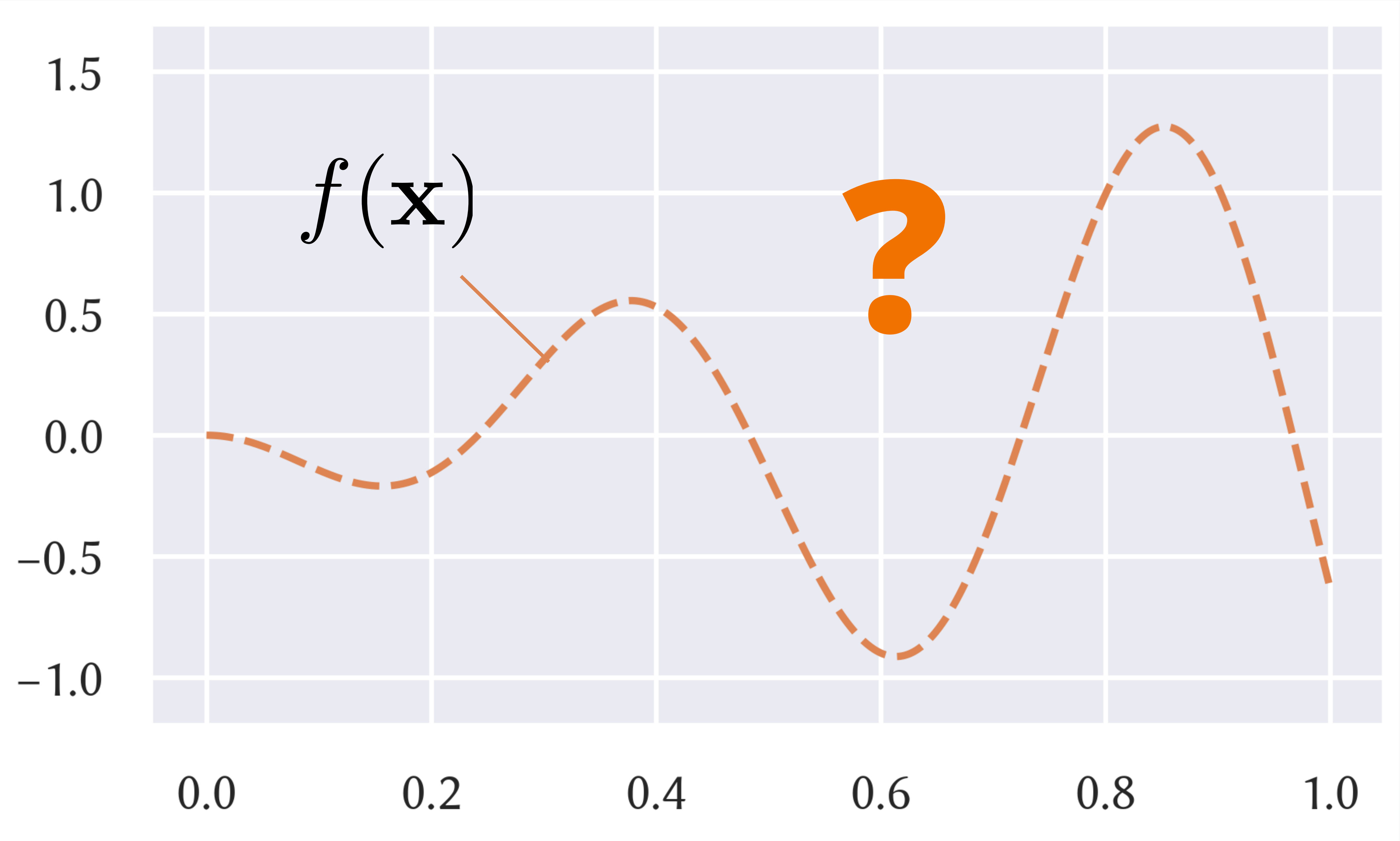
- **Preference Bayesian optimization (PBO)** is a promising technique for human-in-the-loop optimization frameworks
  - PBO is an extension of Bayesian optimization (BO) tailored for human-in-the-loop settings

An Approach: Preferential Bayesian Optimization

- **Preference Bayesian optimization (PBO)** is a promising technique for human-in-the-loop optimization frameworks
  - PBO is an extension of Bayesian optimization (BO) tailored for human-in-the-loop settings
- **Benefits**
  - PBO can proceed using **relative assessment** queries (instead of absolute scoring)
  - PBO can find a good solution with **only a small number of queries**

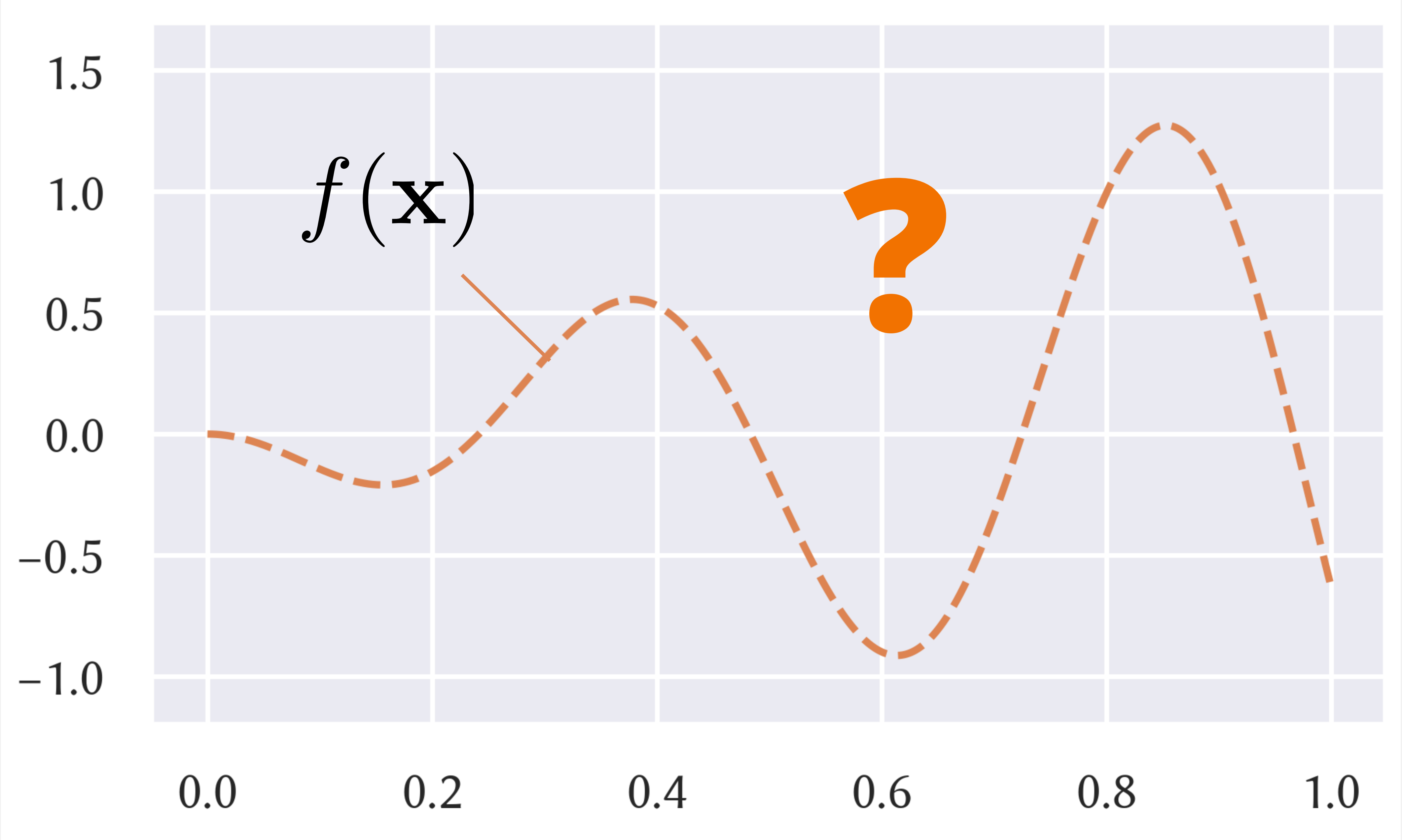
What is Bayesian Optimization?

Problem Setting



$$\mathbf{x}^* = \underset{\mathbf{x} \in \mathbb{R}^n}{\arg\max}\, f(\mathbf{x})$$

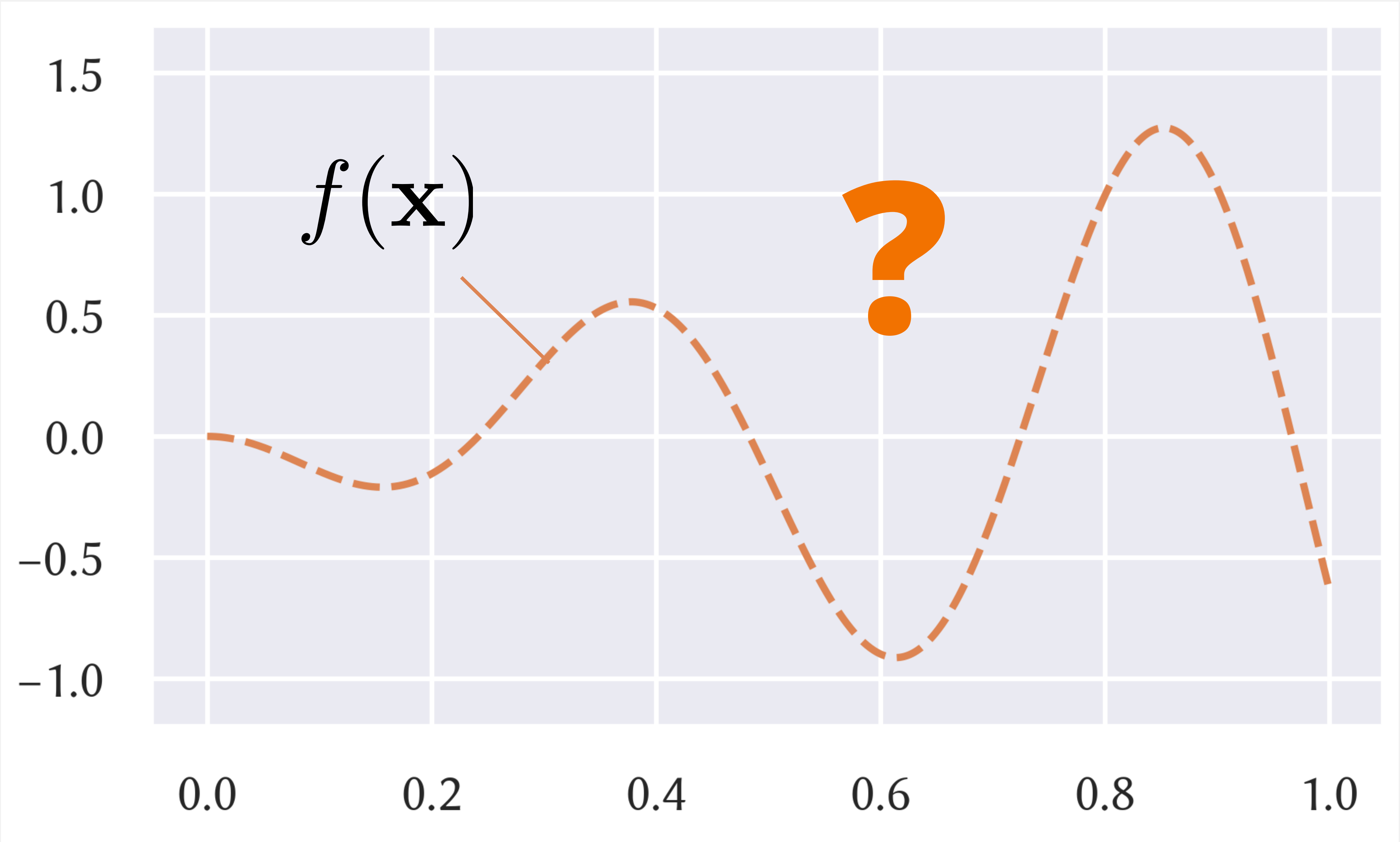
Problem Setting



$$\mathbf{x}^* = \underset{\mathbf{x}\in\mathbb{R}^n}{\arg\max}\, f(\mathbf{x})$$

- A maximization problem with a **"black-box"** objective function
  - We cannot observe the gradient
  - The function can have multiple local maxima

Problem Setting

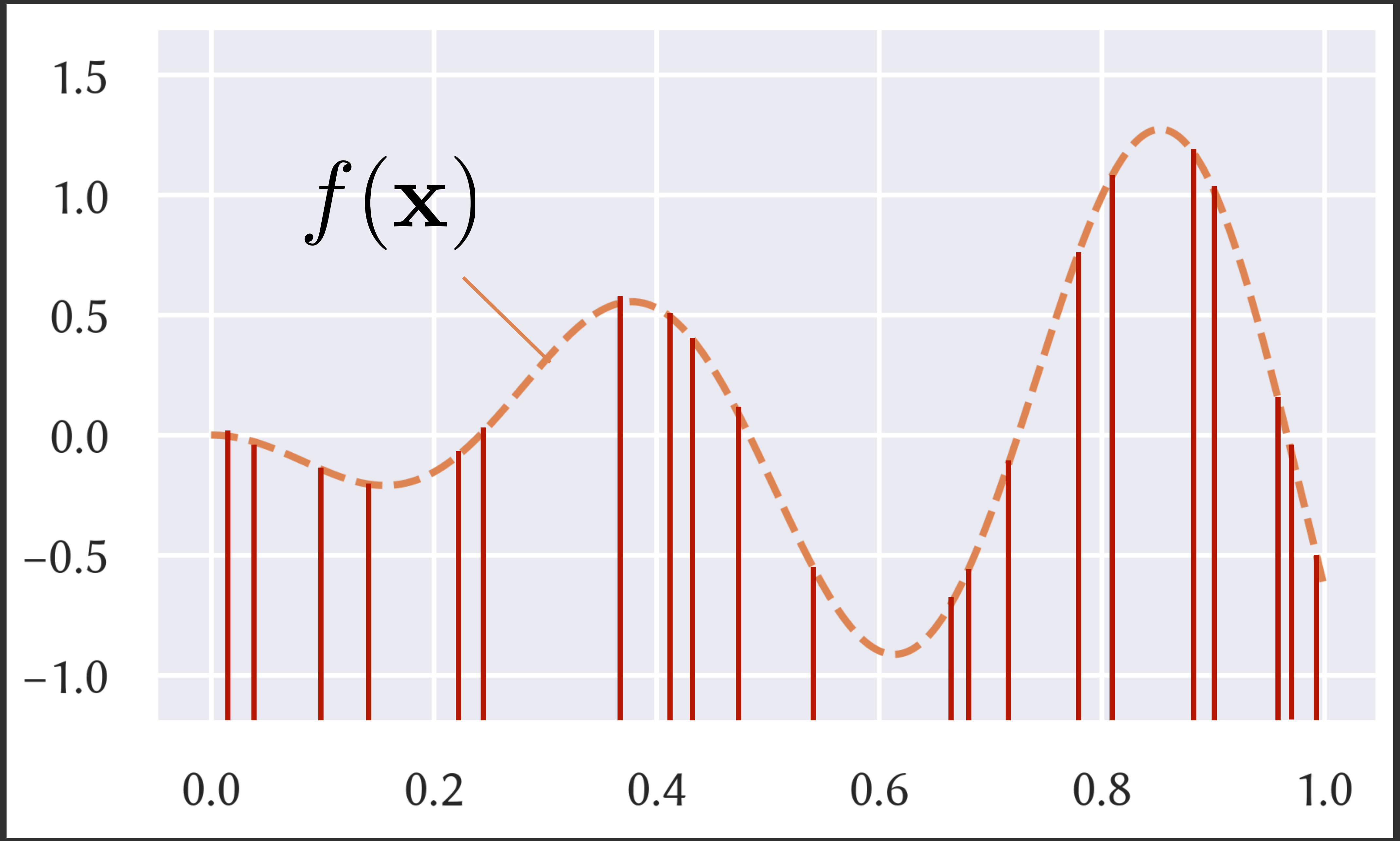


$$\mathbf{x}^* = \arg\max_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$$

- A maximization problem with a **"black-box"** objective function
  - We cannot observe the gradient
  - The function can have multiple local maxima
- **Q: How should we sample points?**

Problem Setting

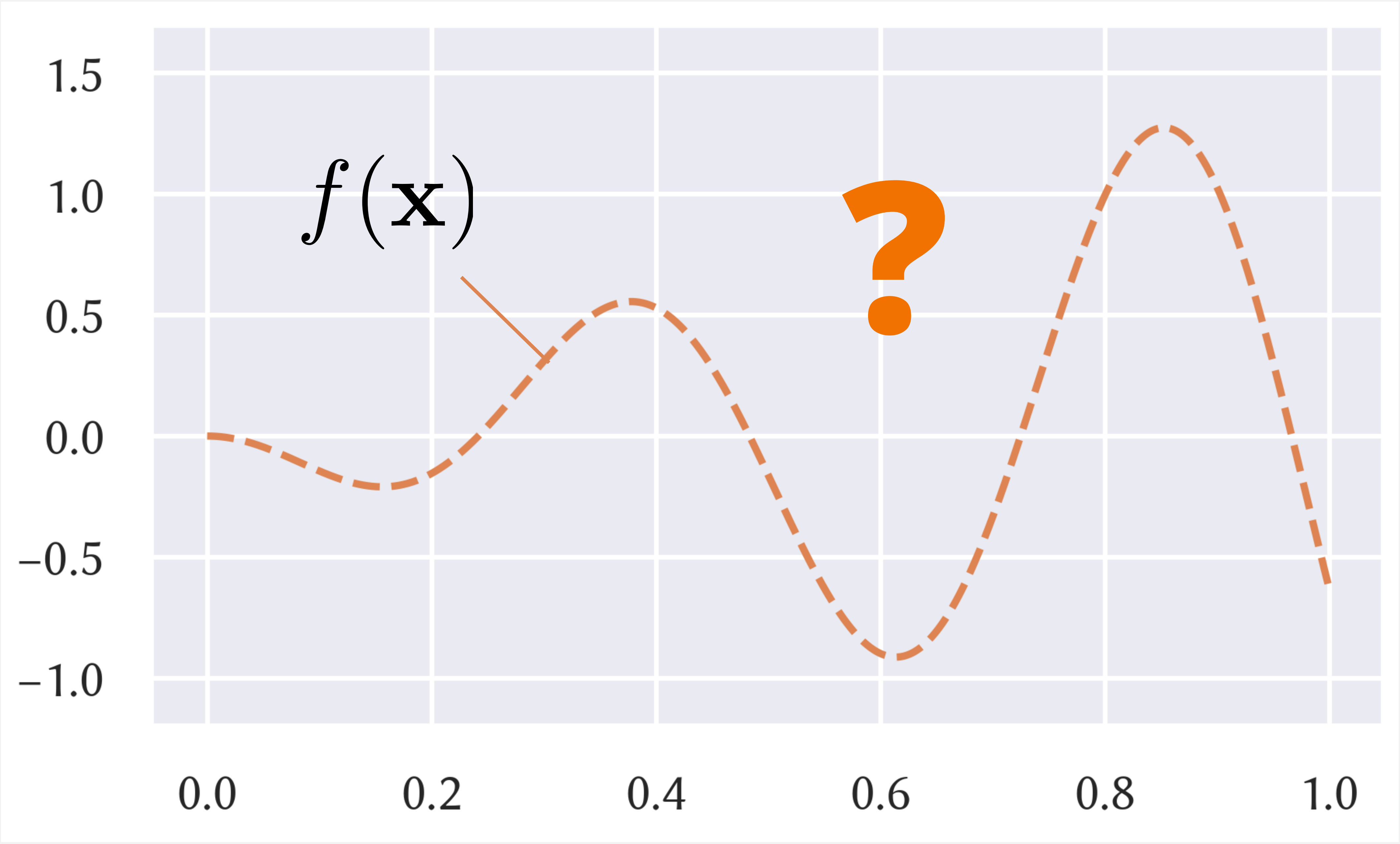
A naïve strategy: **random sampling**



$$\mathbf{x}^* = \underset{\mathbf{x} \in \mathbb{R}^n}{\arg\max} \, f(\mathbf{x})$$

- A maximization problem with a **"black-box"** objective function
  - We cannot observe the gradient
  - The function can have multiple local maxima
- **Q: How should we sample points?**

Problem Setting



$$\mathbf{x}^* = \underset{\mathbf{x} \in \mathbb{R}^n}{\arg\max} \, f(\mathbf{x})$$

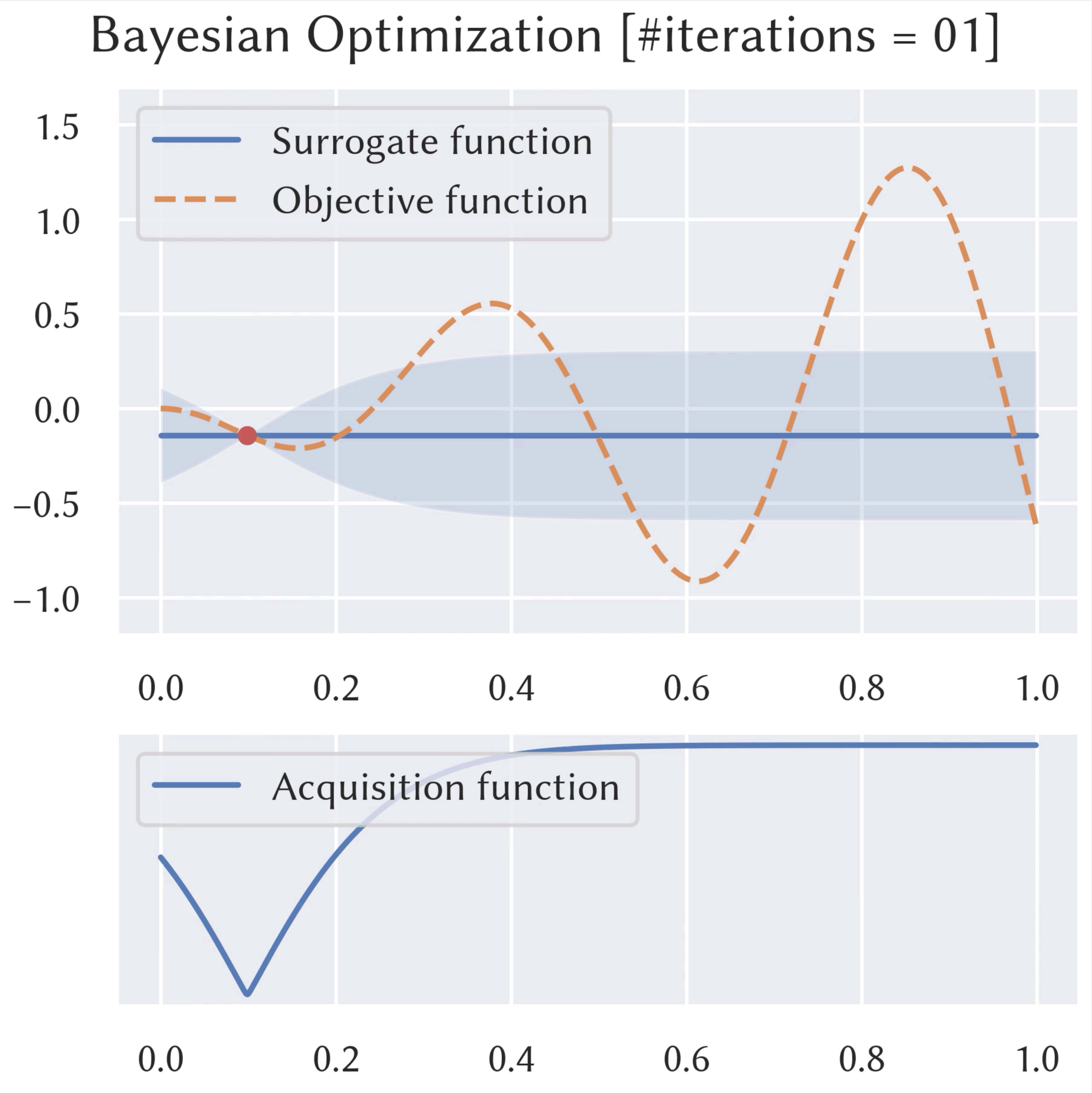
- A maximization problem with a **"black-box"** objective function
  - We cannot observe the gradient
  - The function can have multiple local maxima
- **Q: How should we sample points?**
  - We want to minimize the number of necessary sampling points

Bayesian Optimization (BO)

B. Shahriari, K. Swersky, Z. Wang, R. P. Adams and N. de Freitas, "Taking the Human Out of the Loop: A Review of Bayesian Optimization," in Proceedings of the IEEE, vol. 104, no. 1, pp. 148-175, Jan. 2016, doi: 10.1109/JPROC.2015.2494218

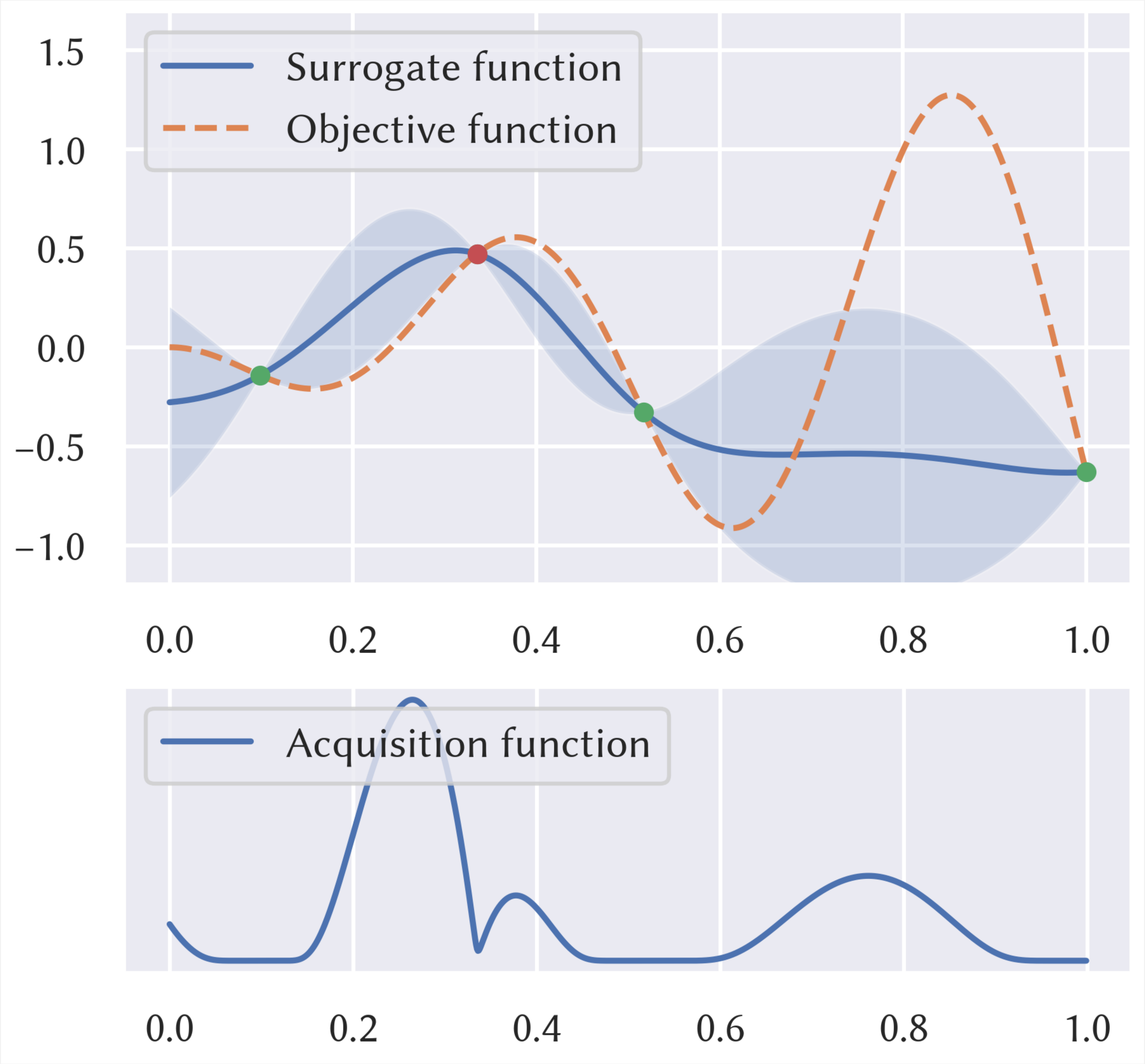
Bayesian Optimization (BO)

- BO is a global “black-box” optimization algorithm
- BO can find optimal solutions with **only a small number of function evaluations** by the use of “acquisition function” (next slide)
  - Thus, BO is useful especially when the objective function is **expensive-to-evaluate**
- Example applications:
  - Hyperparameter tuning for machine learning models [Akiba+, KDD 2019]
  - Fabrication-in-the-loop drawing haptics design [Piovarči+, SIGGRAPH 2020]



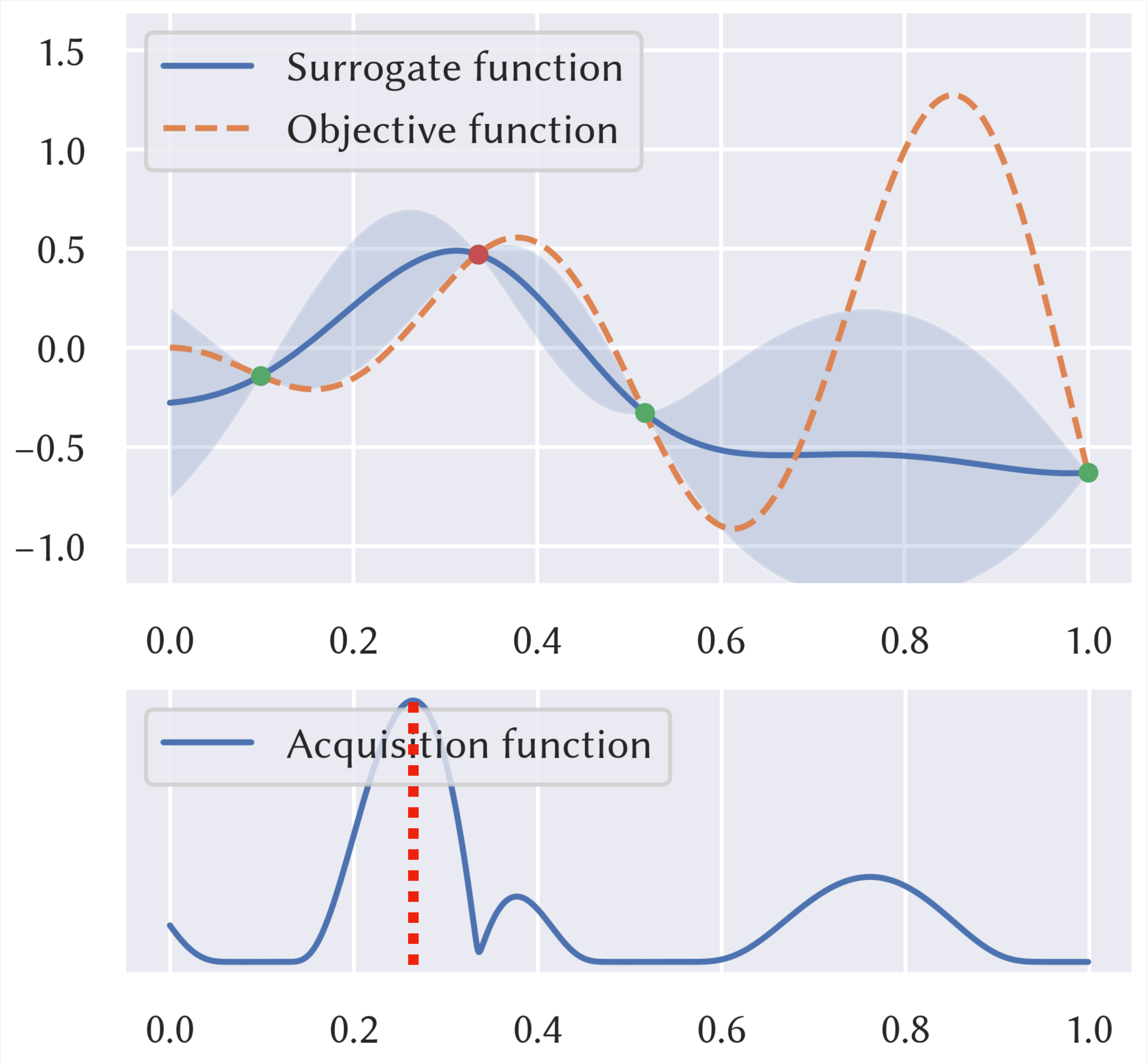
B. Shahriari, K. Swersky, Z. Wang, R. P. Adams and N. de Freitas, "Taking the Human Out of the Loop: A Review of Bayesian Optimization," in Proceedings of the IEEE, vol. 104, no. 1, pp. 148-175, Jan. 2016, doi: 10.1109/JPROC.2015.2494218

Acquisition Function



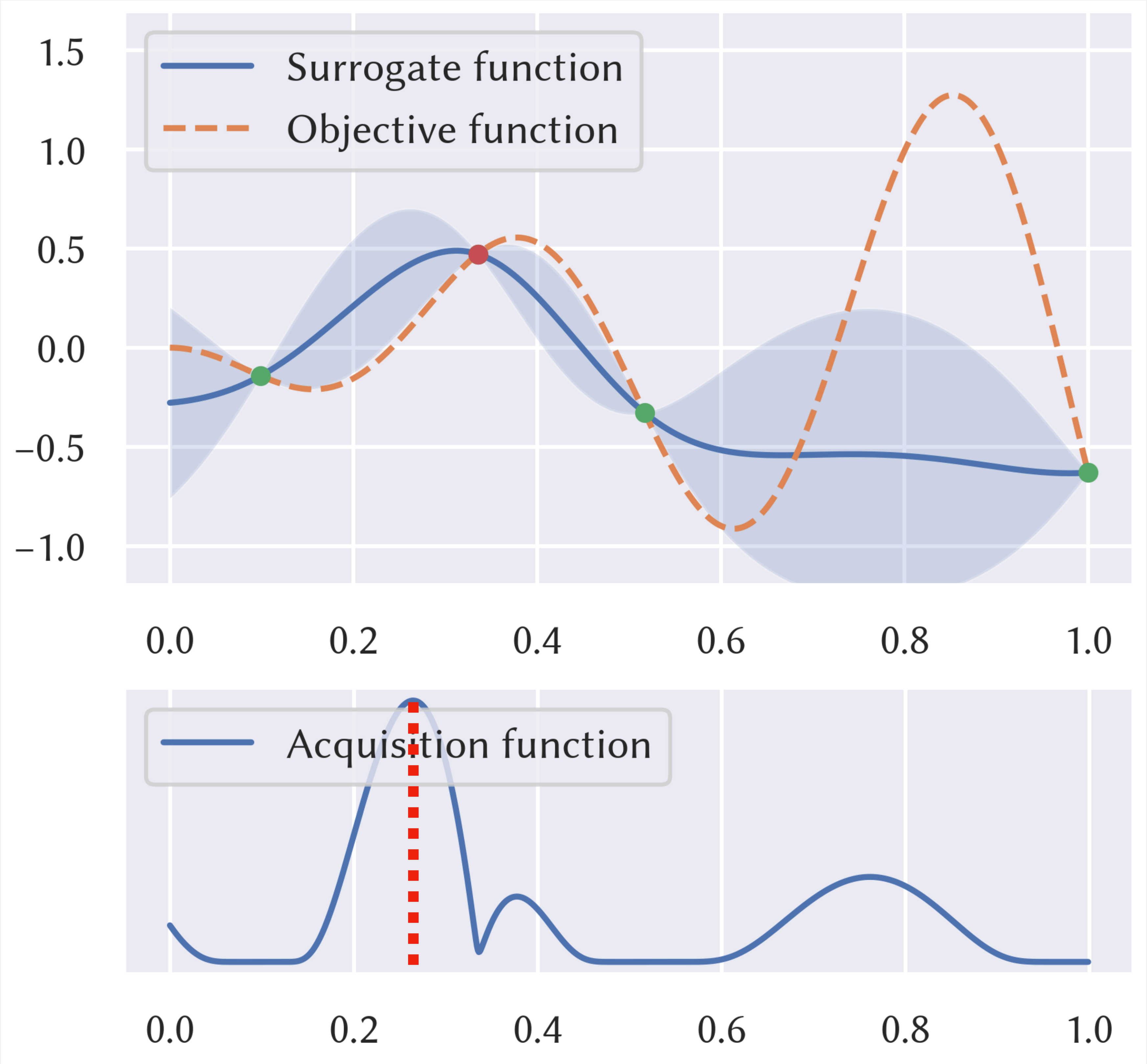
Acquisition Function

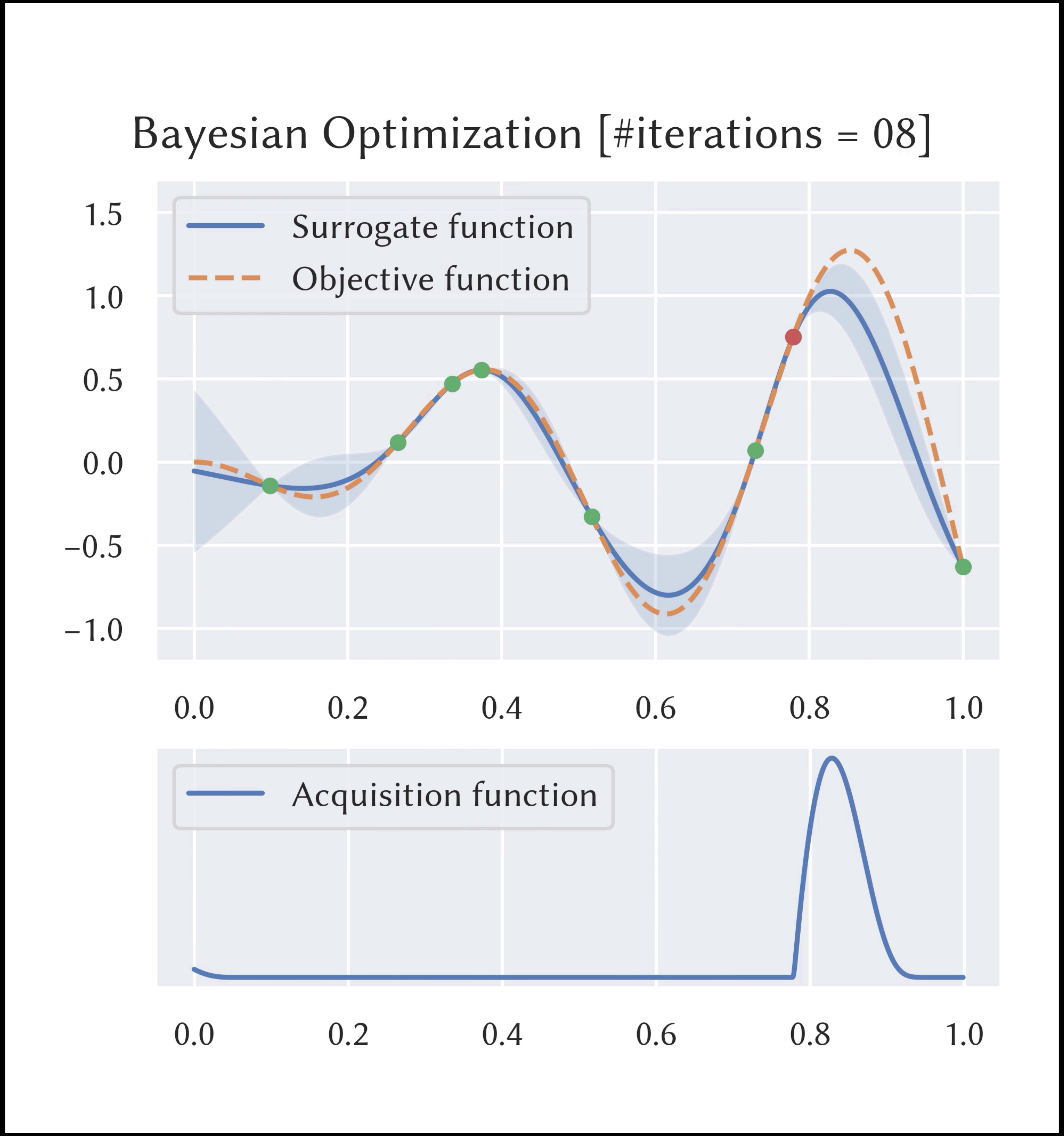
- An acquisition function evaluates **how effective a point is if it is sampled in the next iteration**
  - The maximizer of the acquisition function will be sampled in the next iteration



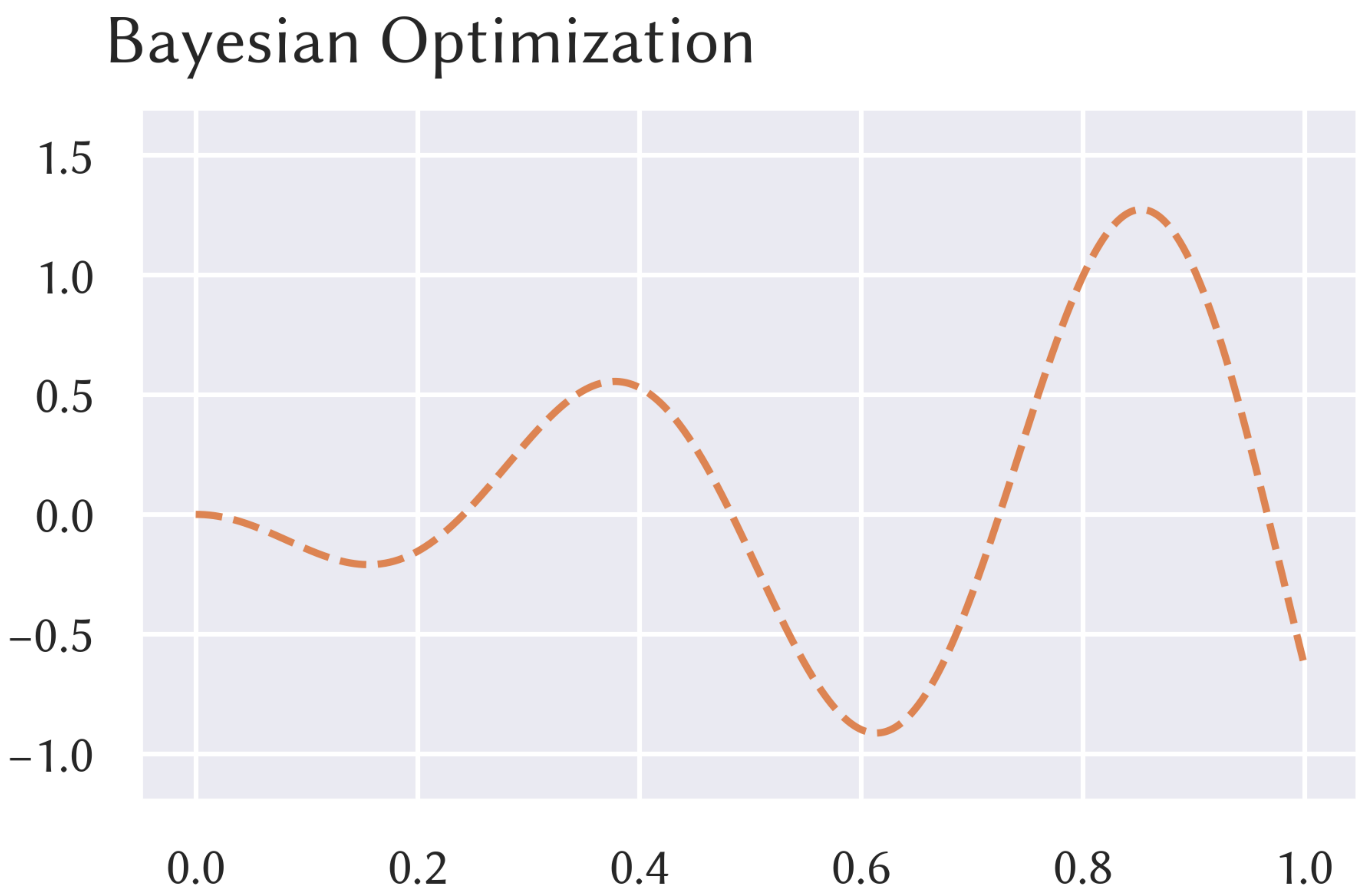
Acquisition Function

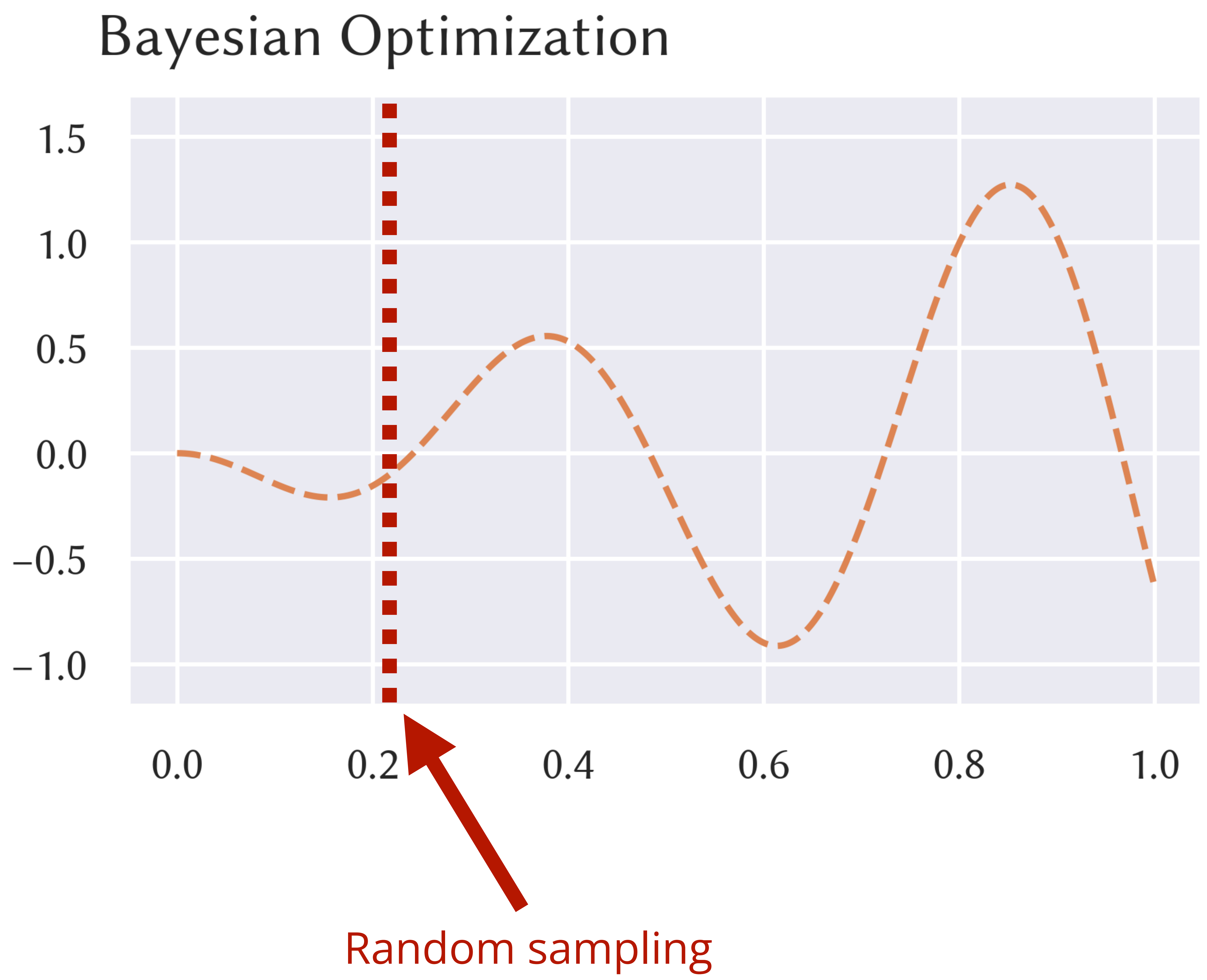
- An acquisition function evaluates **how effective a point is if it is sampled in the next iteration**
    - The maximizer of the acquisition function will be sampled in the next iteration
- Acquisition functions are usually designed to **balance exploration and exploitation**
    - **Exploration:** Prioritize large uncertainty
    - **Exploitation:** Prioritize high predicted value

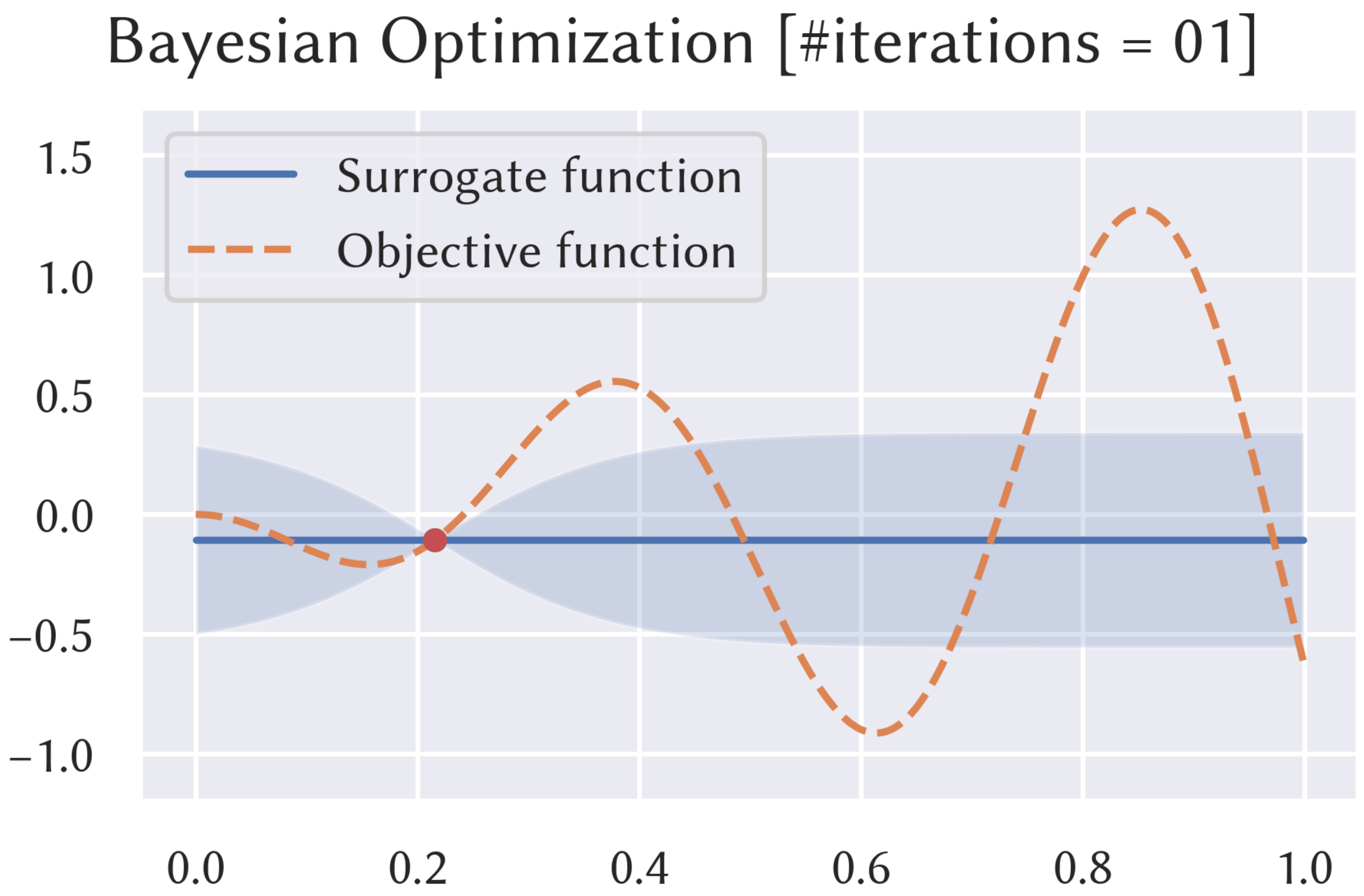


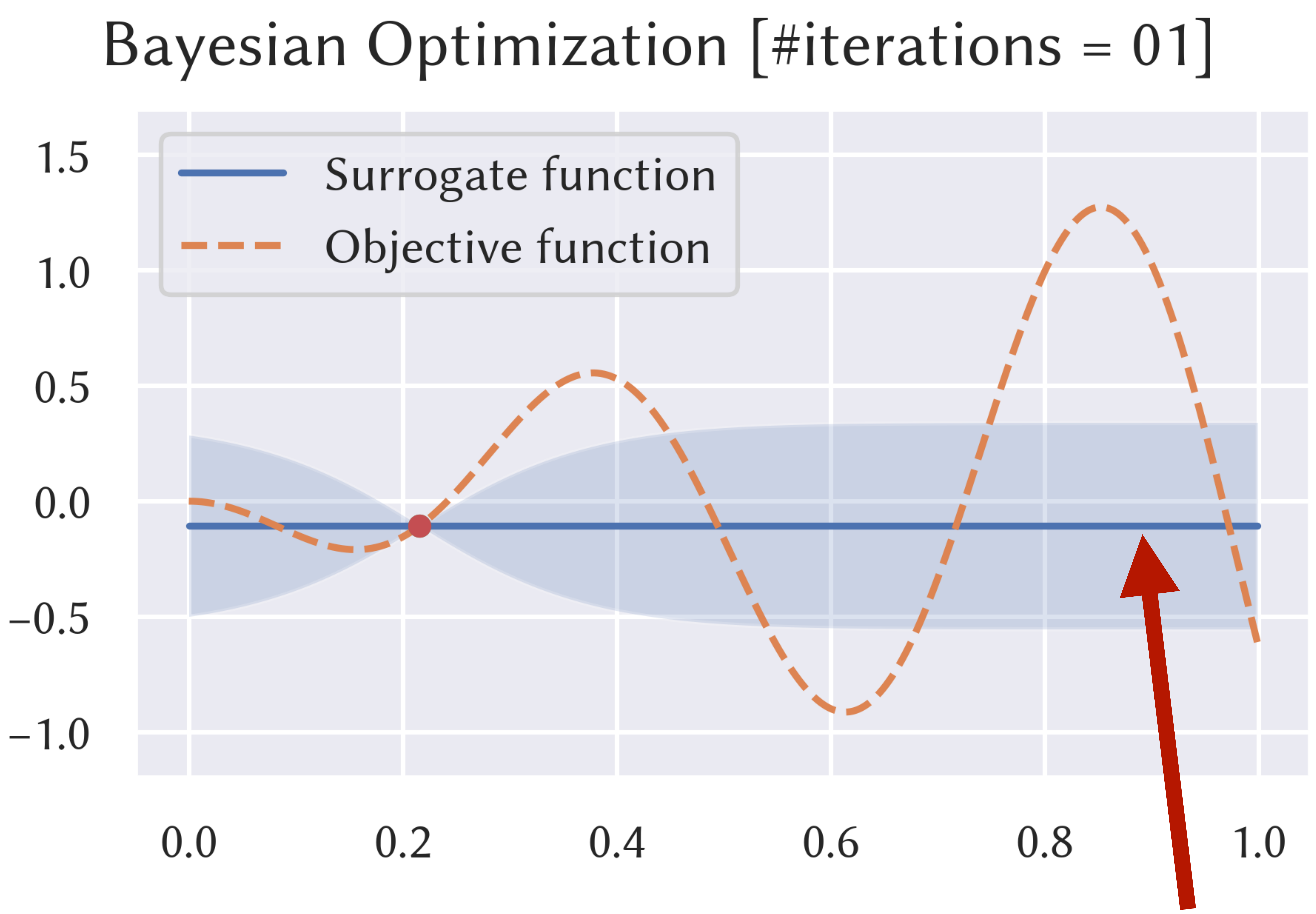

Bayesian Optimization [#iterations = 08]
Surrogate function
Objective function
1.5
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0
Acquisition function
0.0
0.2
0.4
0.6
0.8
1.0

Step-by-Step Explanation

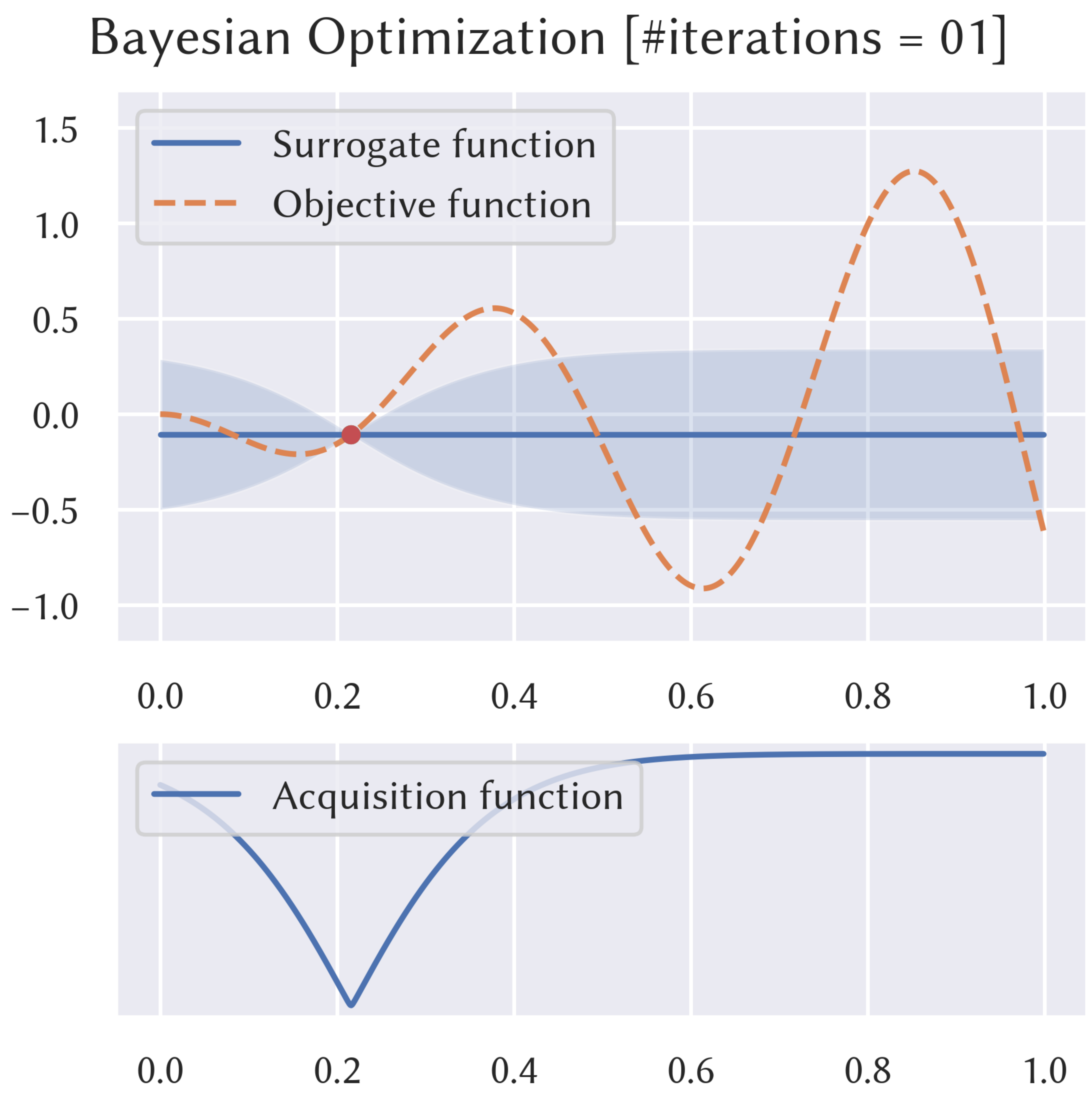

Bayesian Optimization
1.5
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0


Bayesian Optimization
1.5
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0
Random sampling

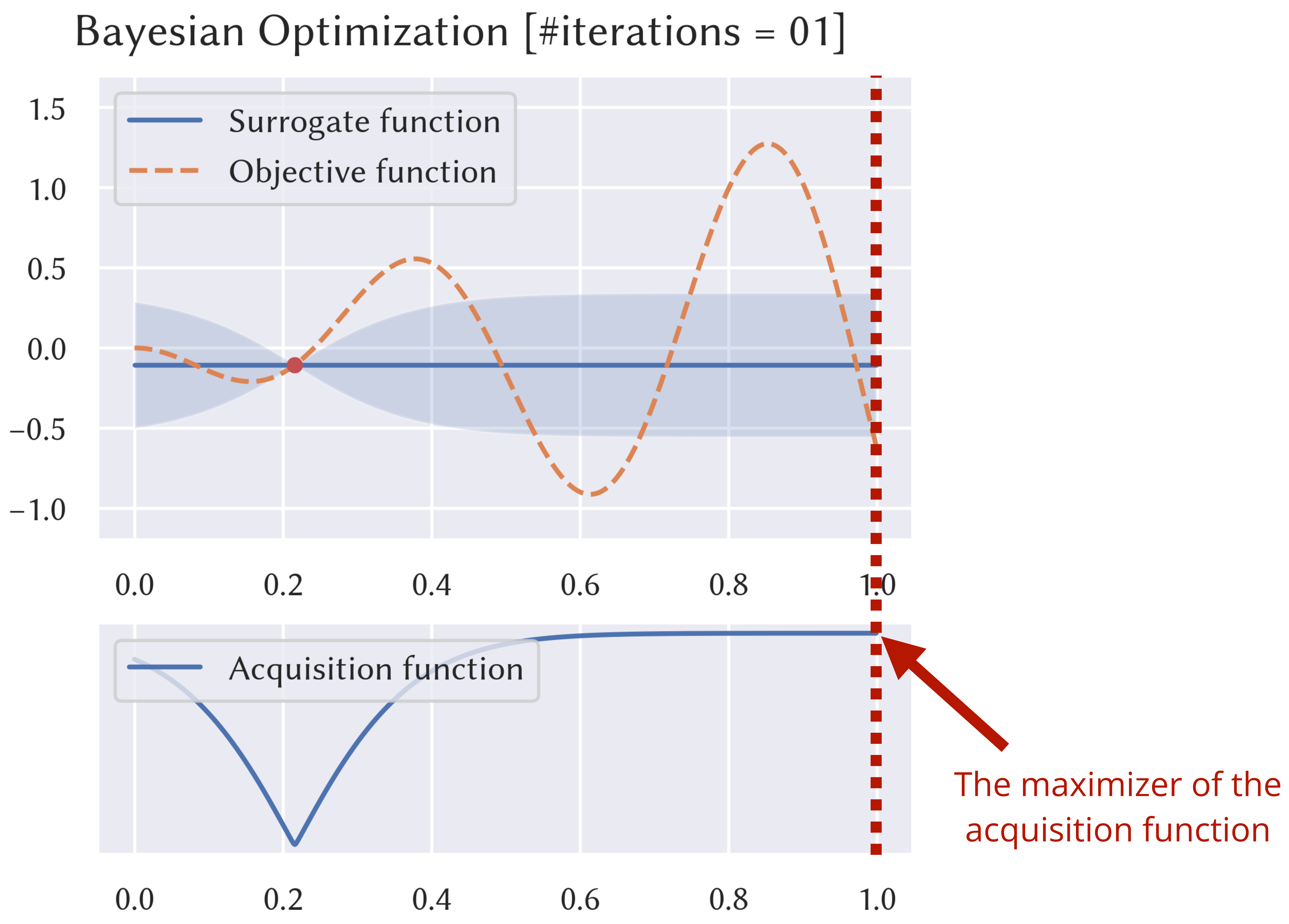

Bayesian Optimization [#iterations = 01]
Surrogate function
Objective function
1.5
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0

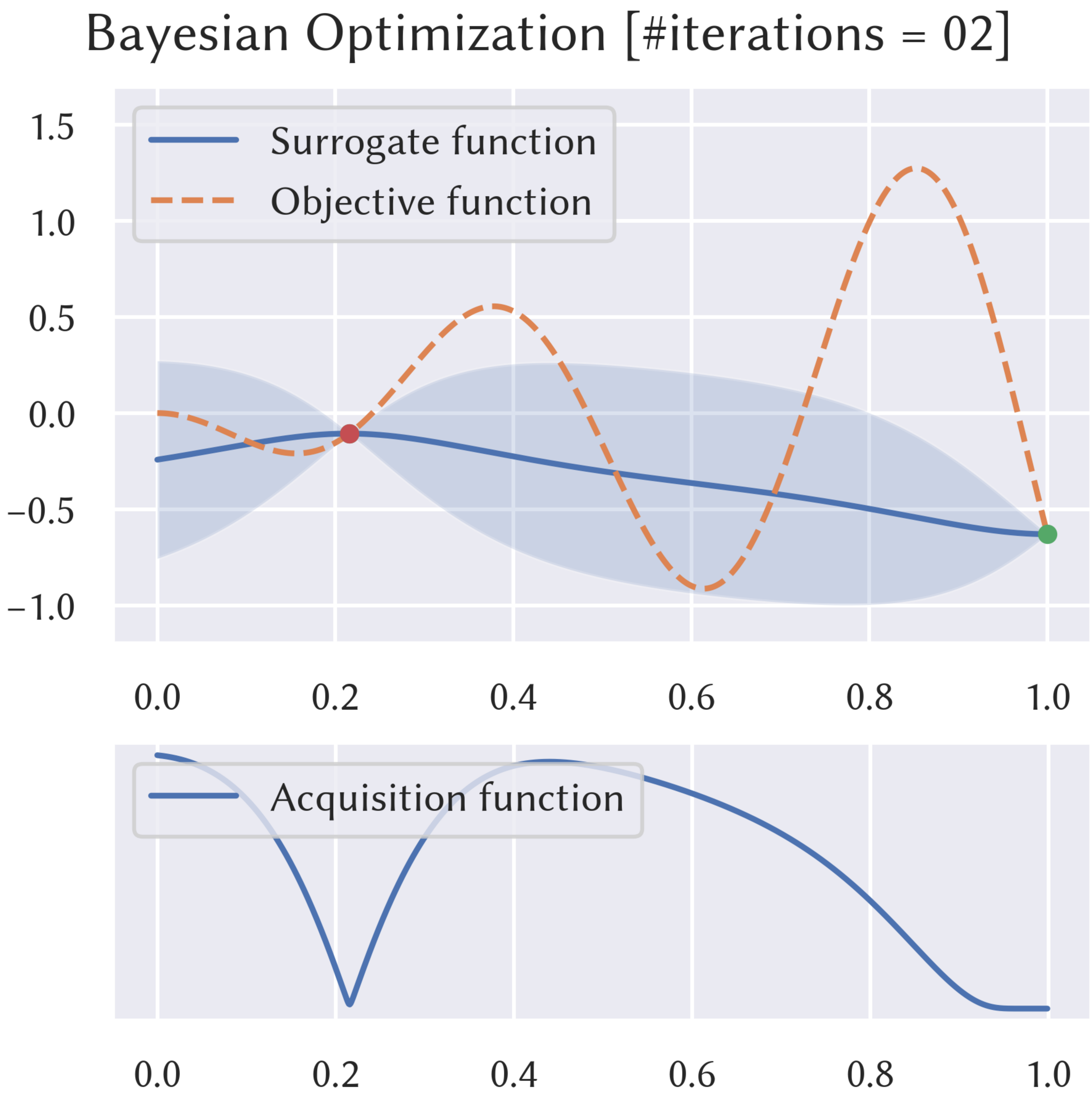


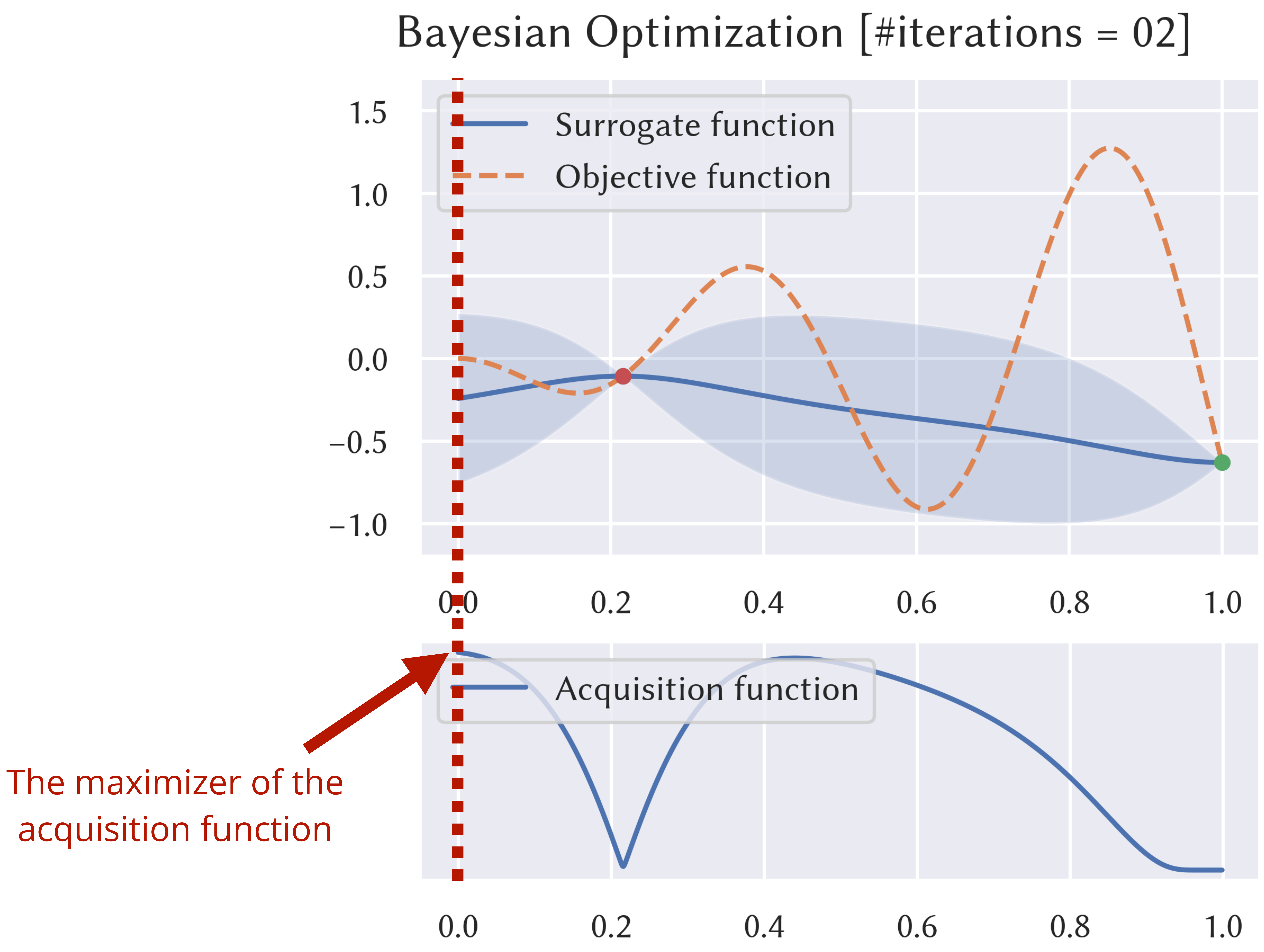
Gaussian process regression
(The light blue means the confidence)



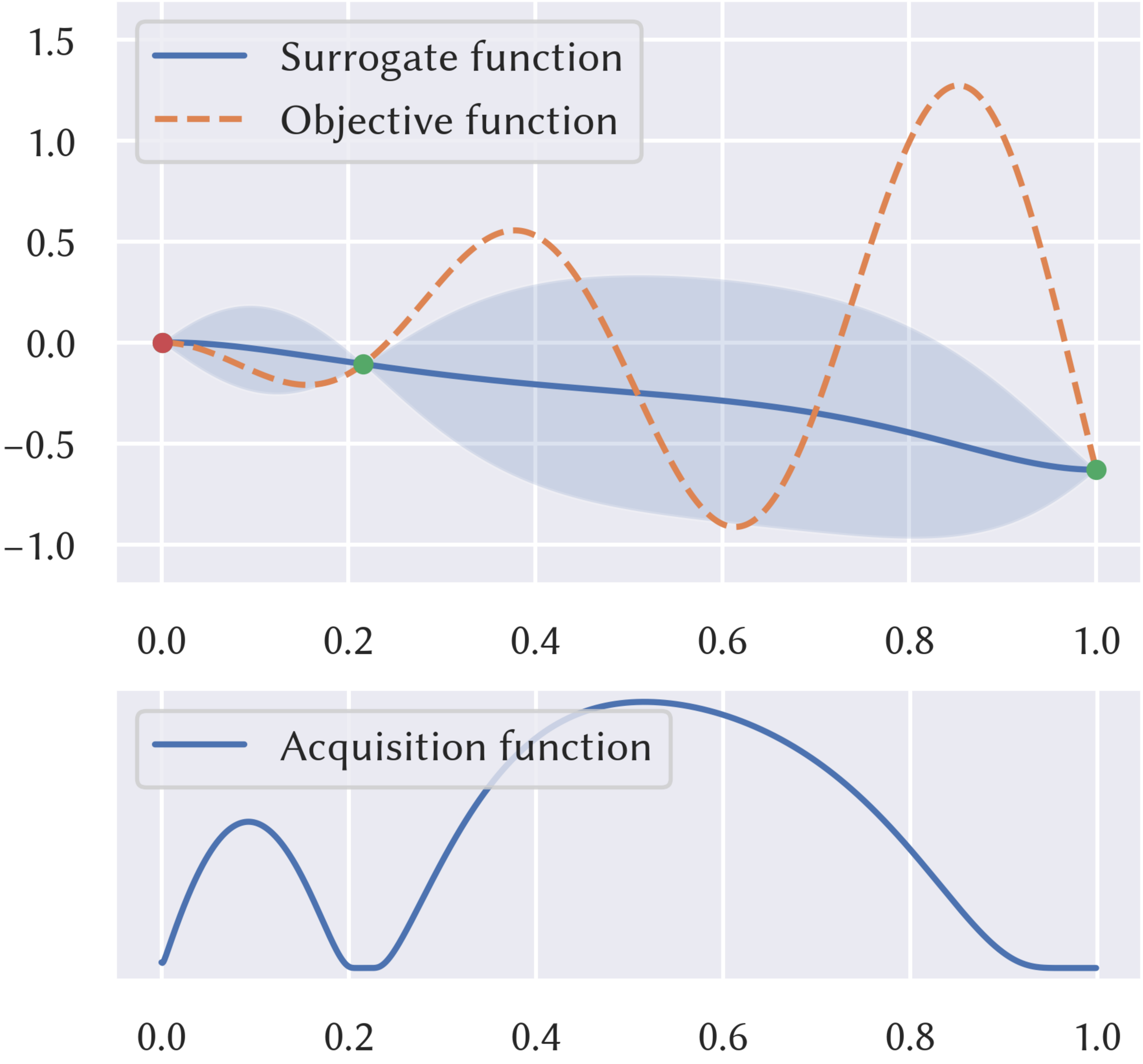
Bayesian Optimization [#iterations = 01]
Surrogate function
Objective function
1.5
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0
Acquisition function
0.0
0.2
0.4
0.6
0.8
1.0

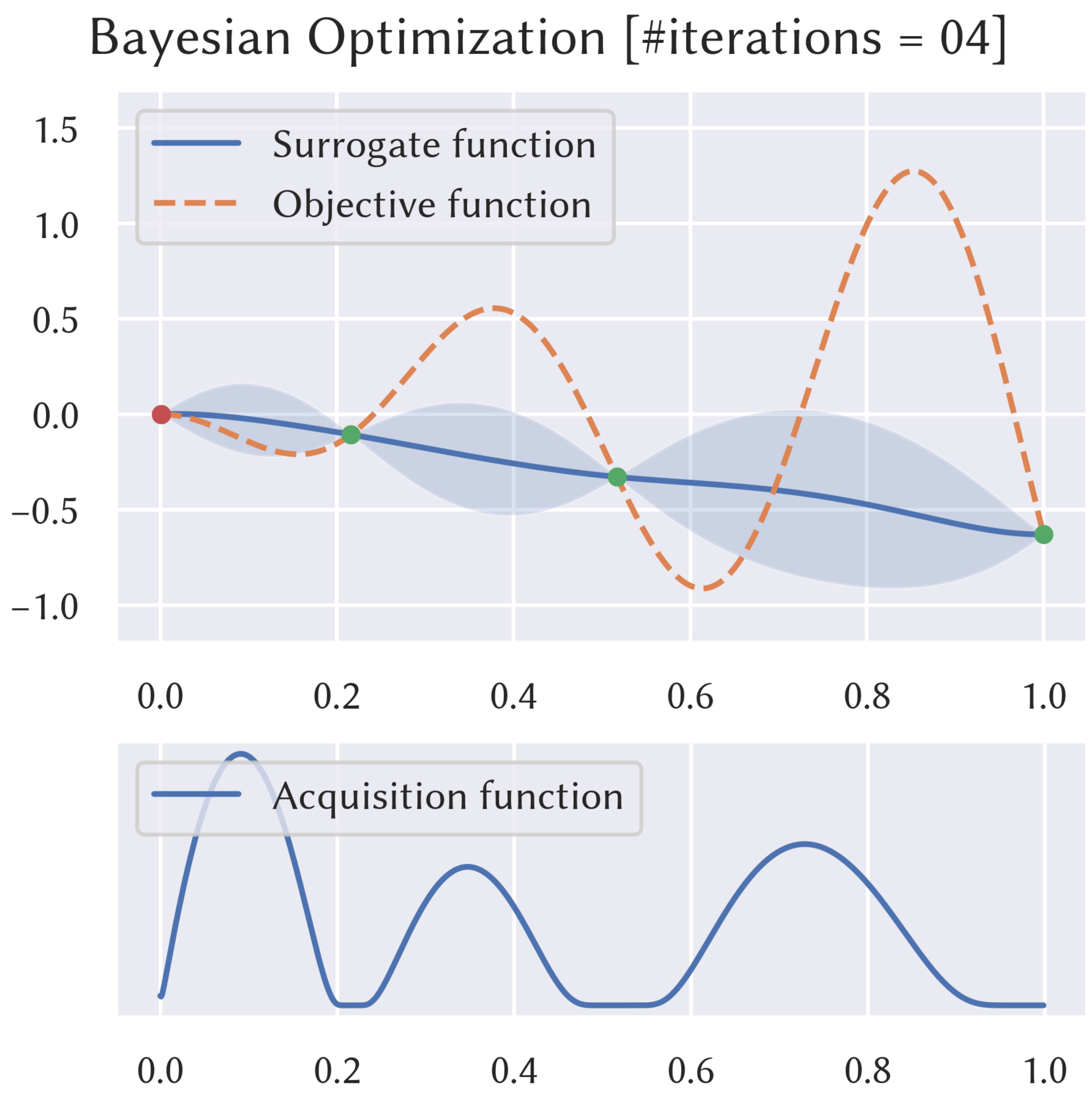

Bayesian Optimization [#iterations = 01]
Surrogate function
Objective function
1.5
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0
Acquisition function
0.0
0.2
0.4
0.6
0.8
1.0
The maximizer of the
acquisition function


Bayesian Optimization [#iterations = 02]
Surrogate function
Objective function
1.5
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0
Acquisition function
0.0
0.2
0.4
0.6
0.8
1.0

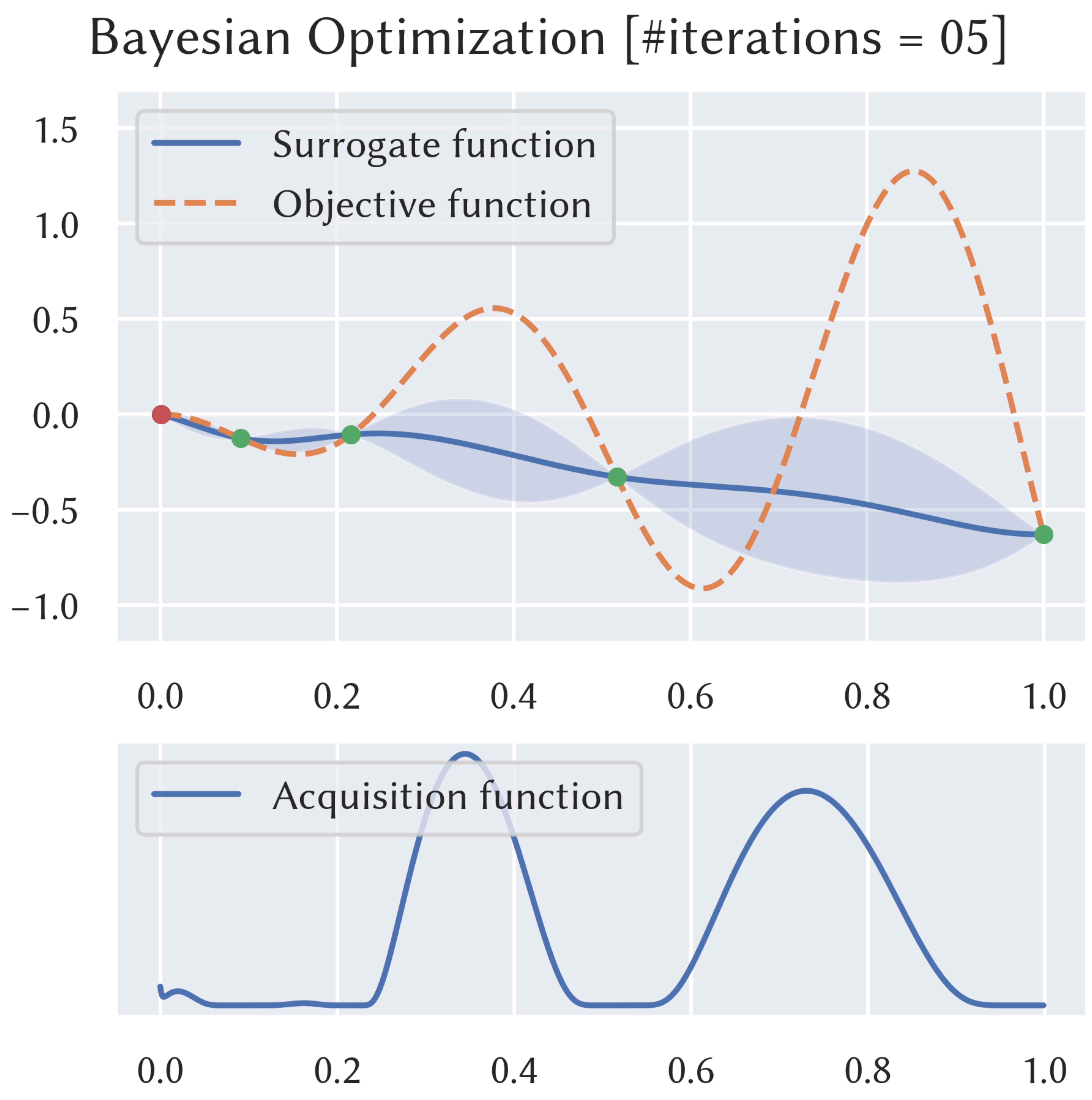

Bayesian Optimization [#iterations = 02]
Surrogate function
Objective function
1.5
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0
Acquisition function
0.0
0.2
0.4
0.6
0.8
1.0
The maximizer of the
acquisition function

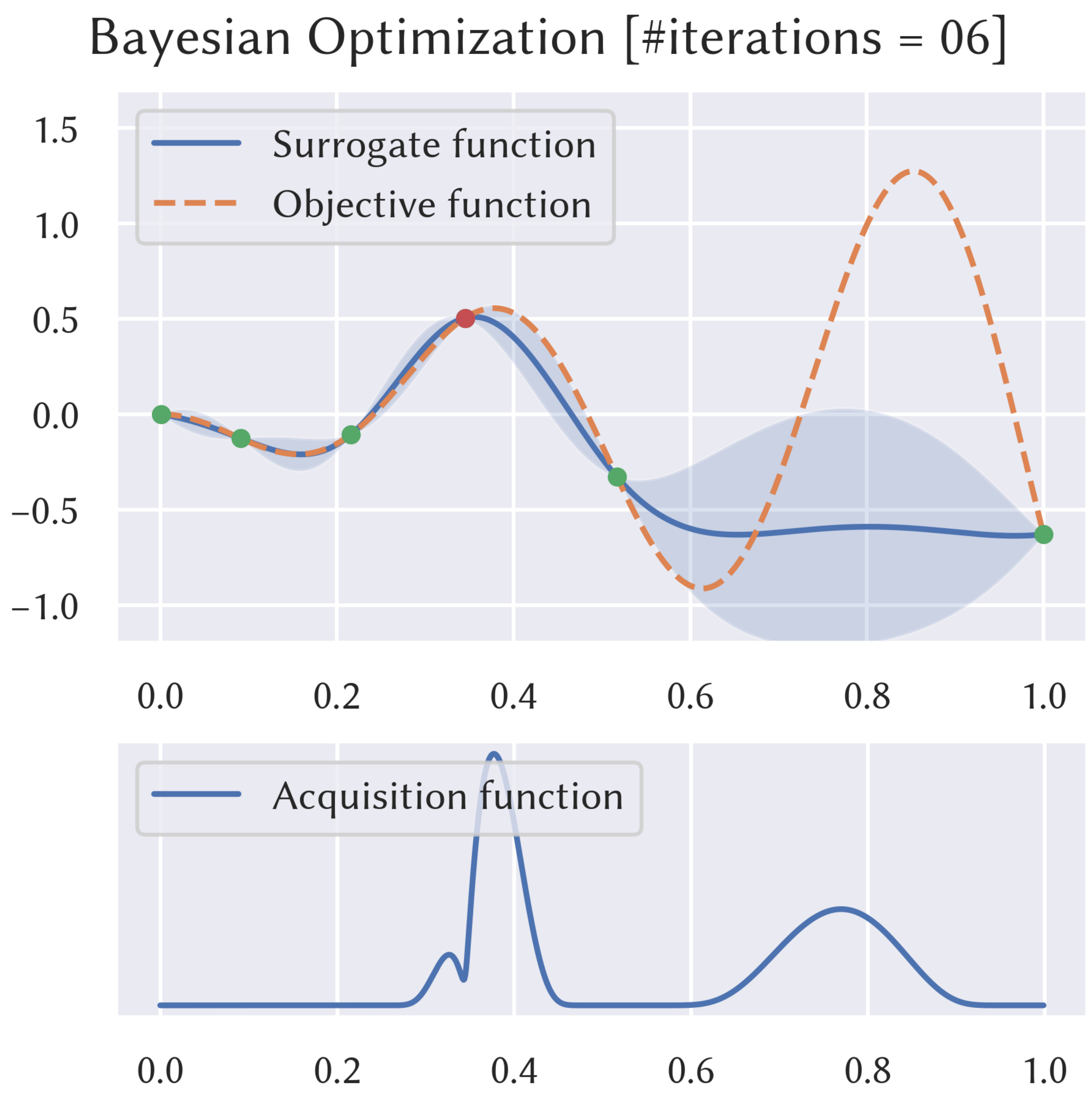
Bayesian Optimization [#iterations = 03]

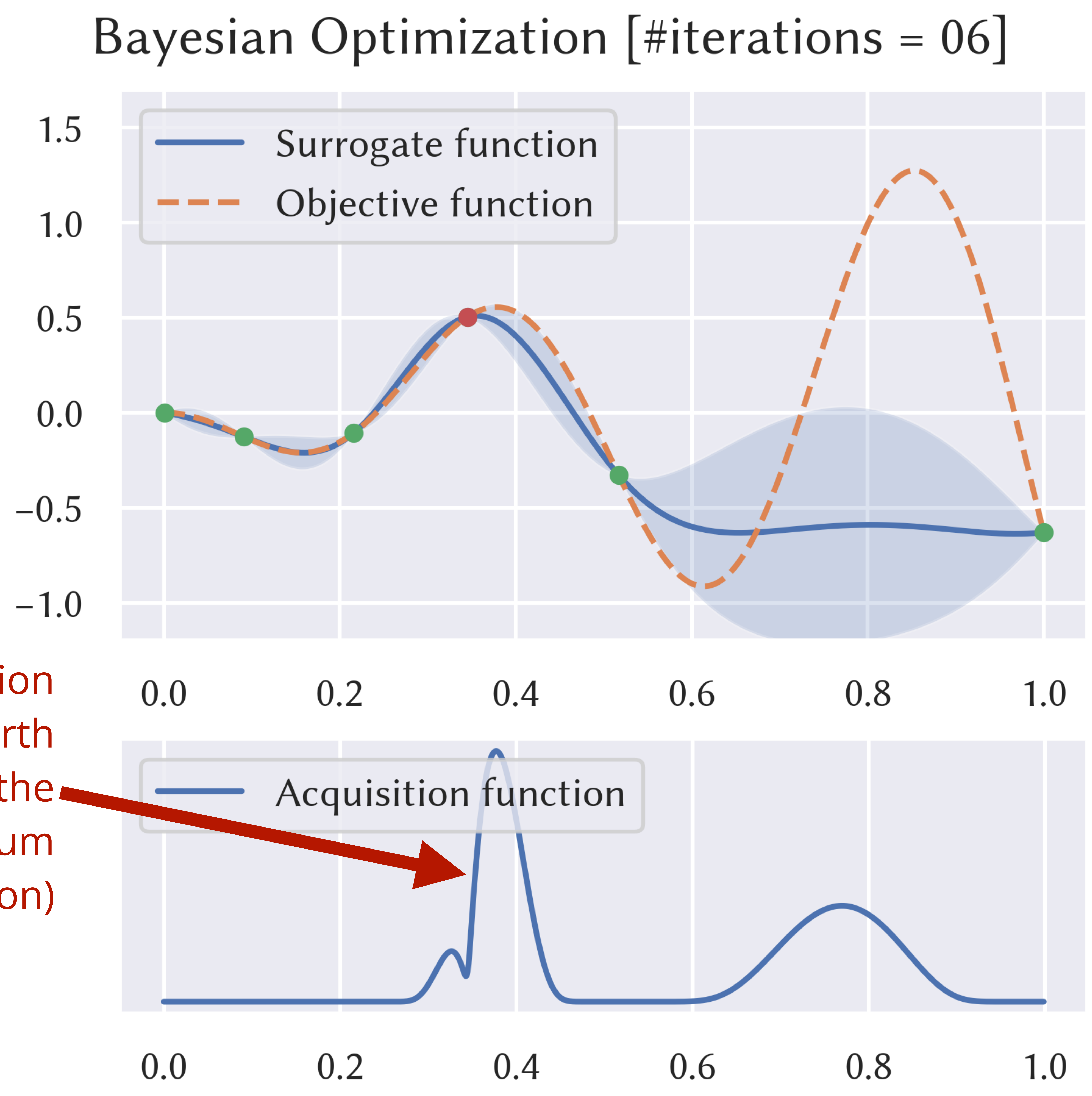




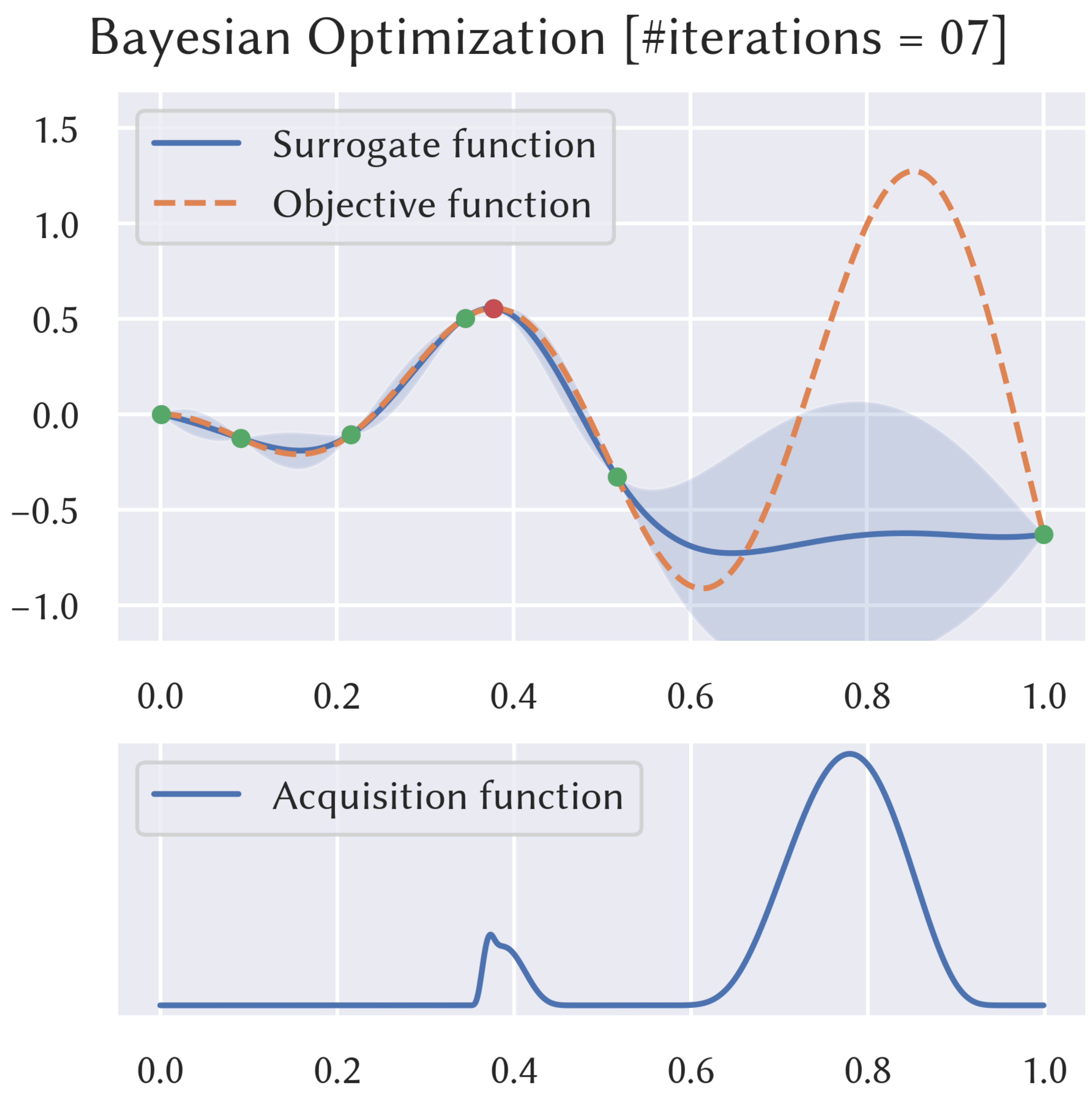
Bayesian Optimization [#iterations = 04]
Surrogate function
Objective function
1.5
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0
Acquisition function
0.0
0.2
0.4
0.6
0.8
1.0


Bayesian Optimization [#iterations = 05]
Surrogate function
Objective function
Acquisition function
1.5
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0

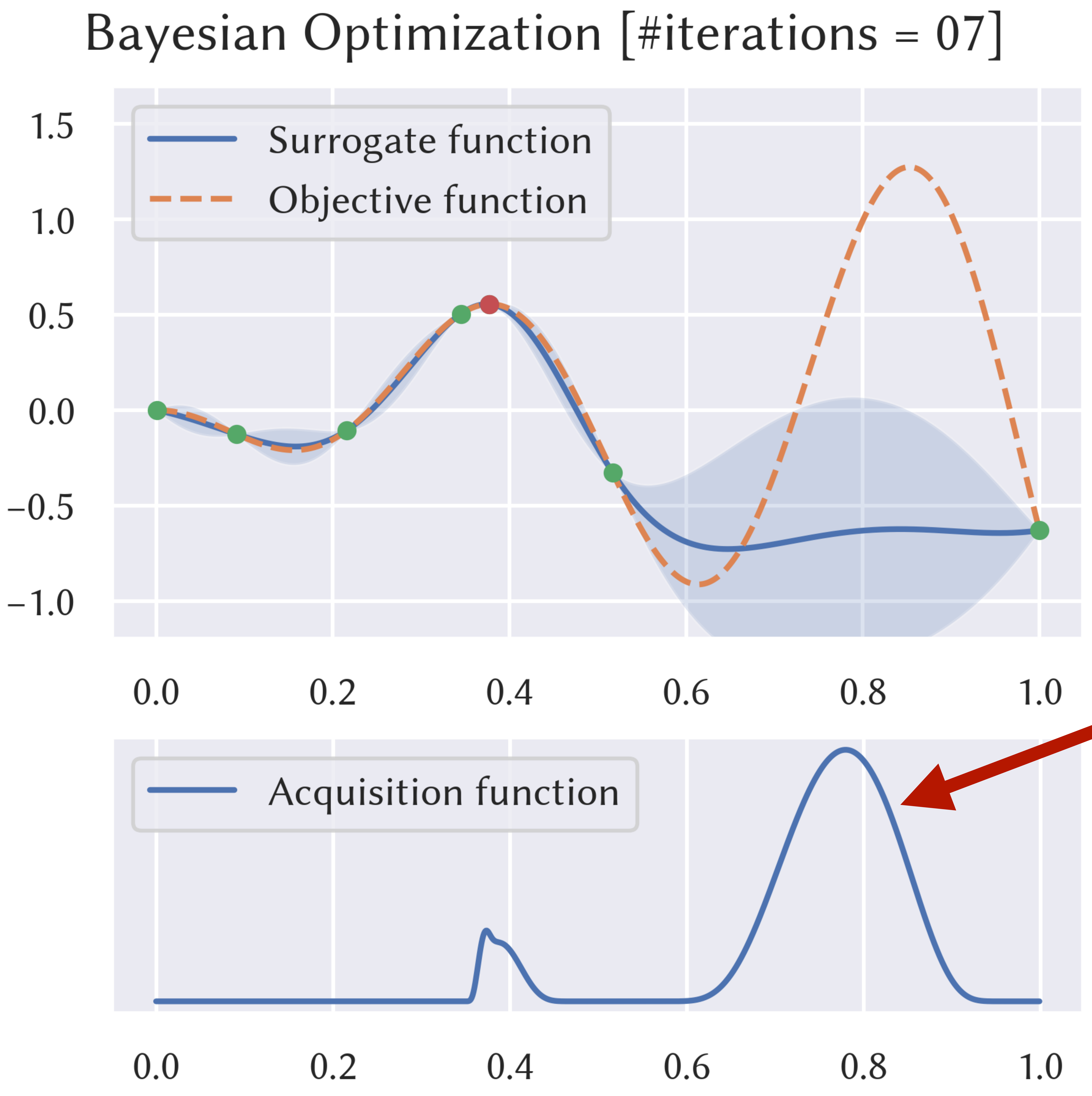

Bayesian Optimization [#iterations = 06]
Surrogate function
Objective function
1.5
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0
Acquisition function
0.0
0.2
0.4
0.6
0.8
1.0



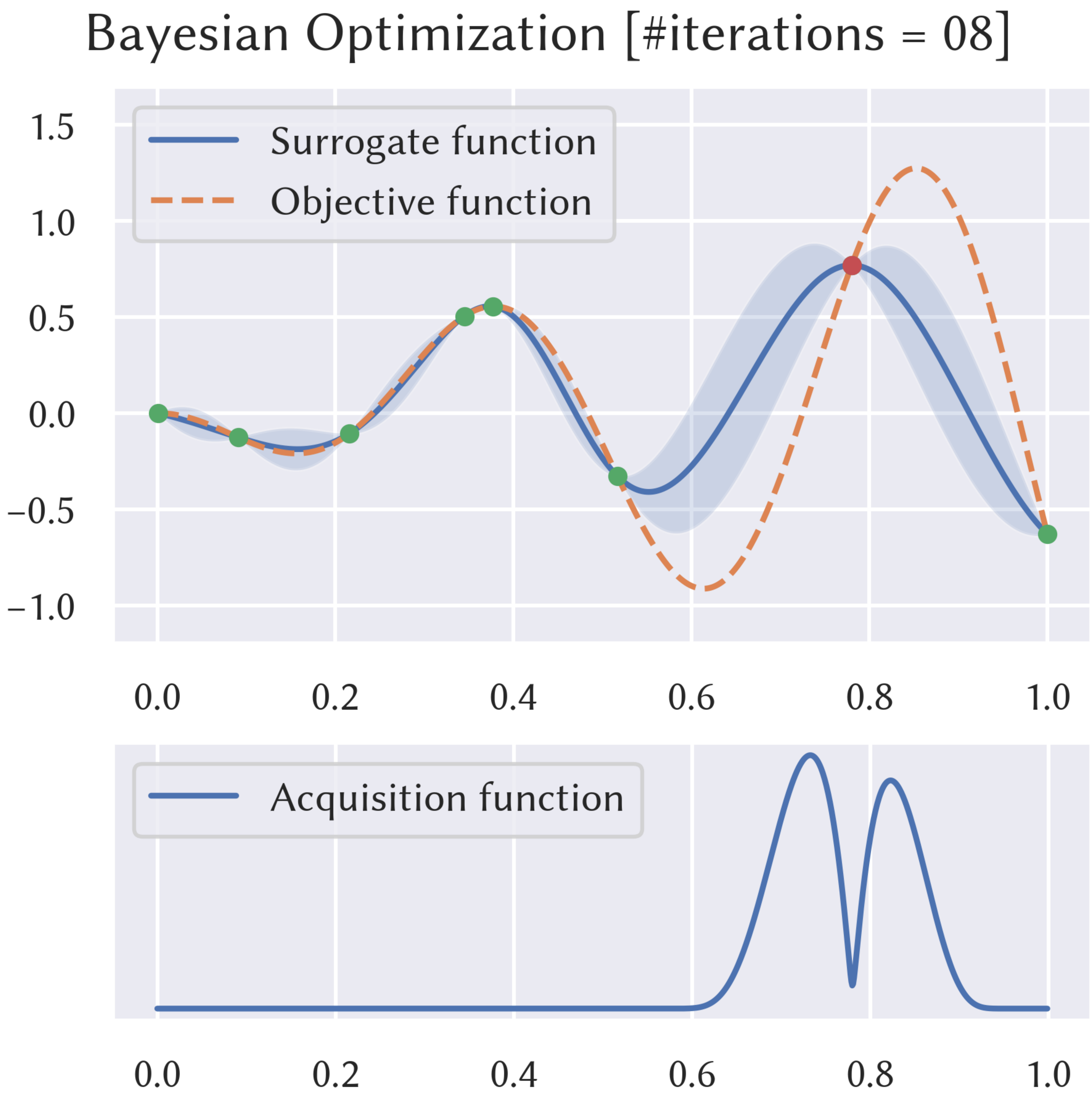
This acquisition function considers it is worth sampling around the current maximum (exploitation)

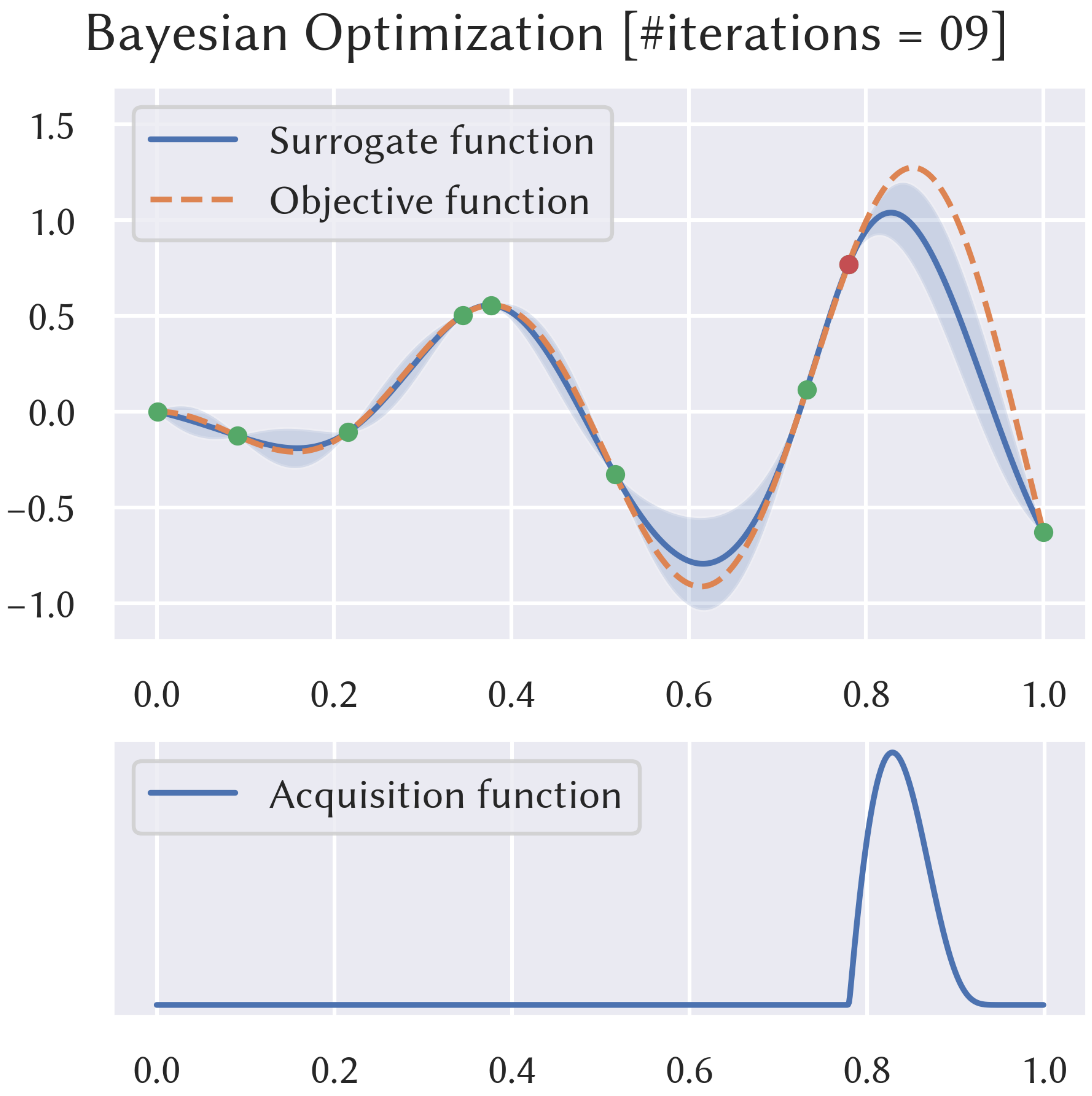


Bayesian Optimization [#iterations = 07]
Surrogate function
Objective function
1.5
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0
Acquisition function
0.0
0.2
0.4
0.6
0.8
1.0

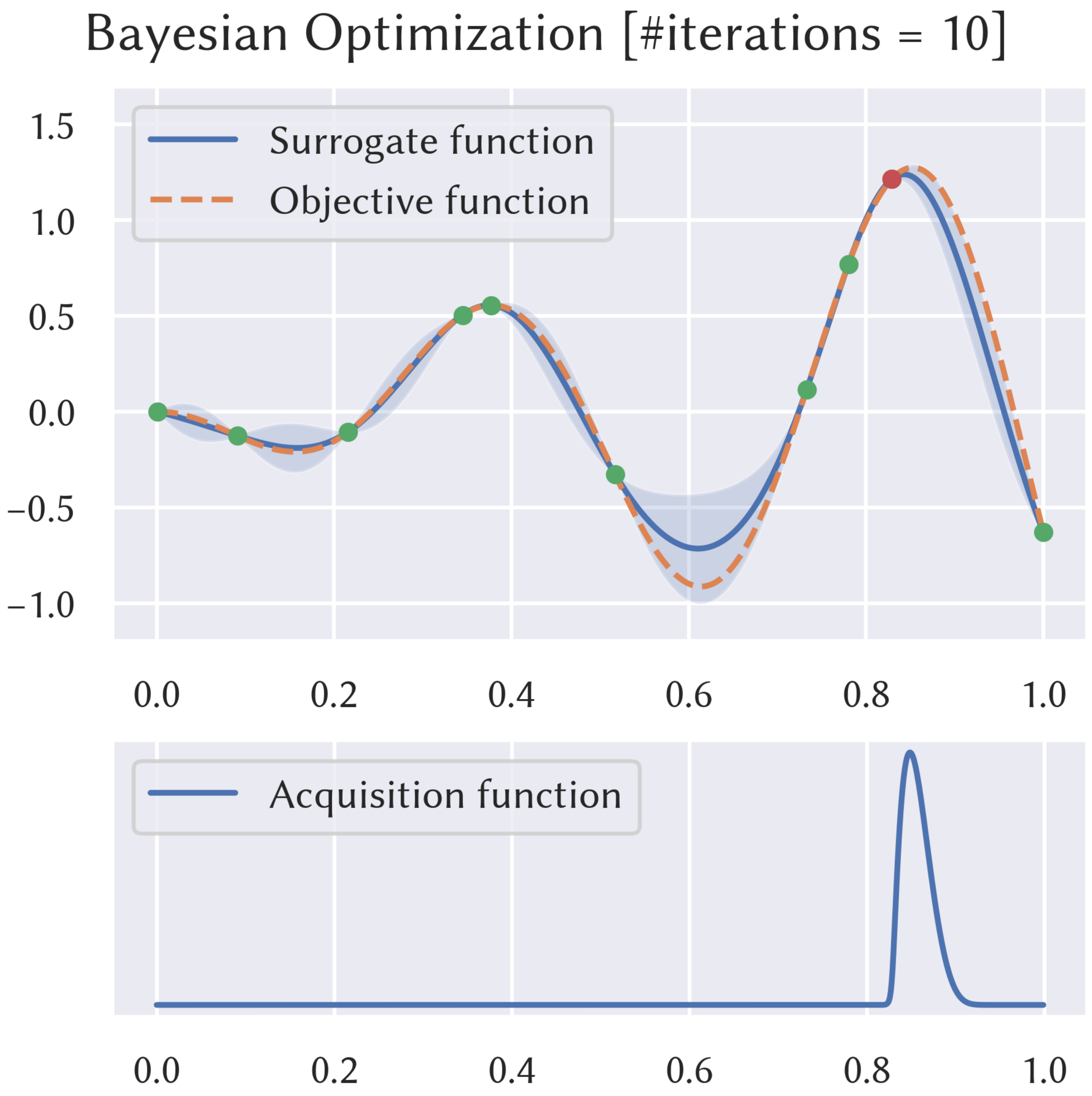


Now this acquisition function considers it is worth sampling the unvisited region (exploration)

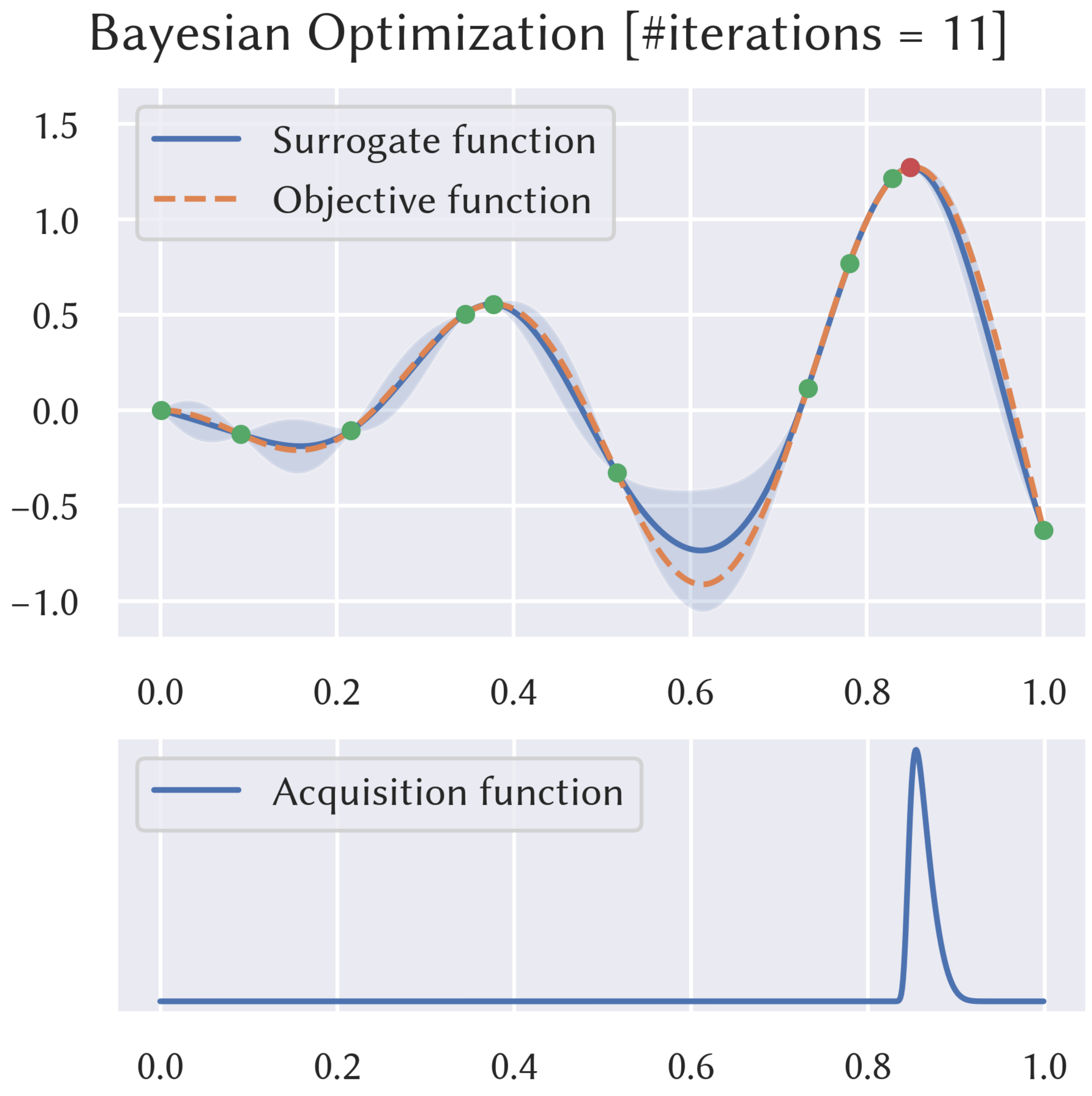

Bayesian Optimization [#iterations = 08]
Surrogate function
Objective function
Acquisition function
1.5
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0

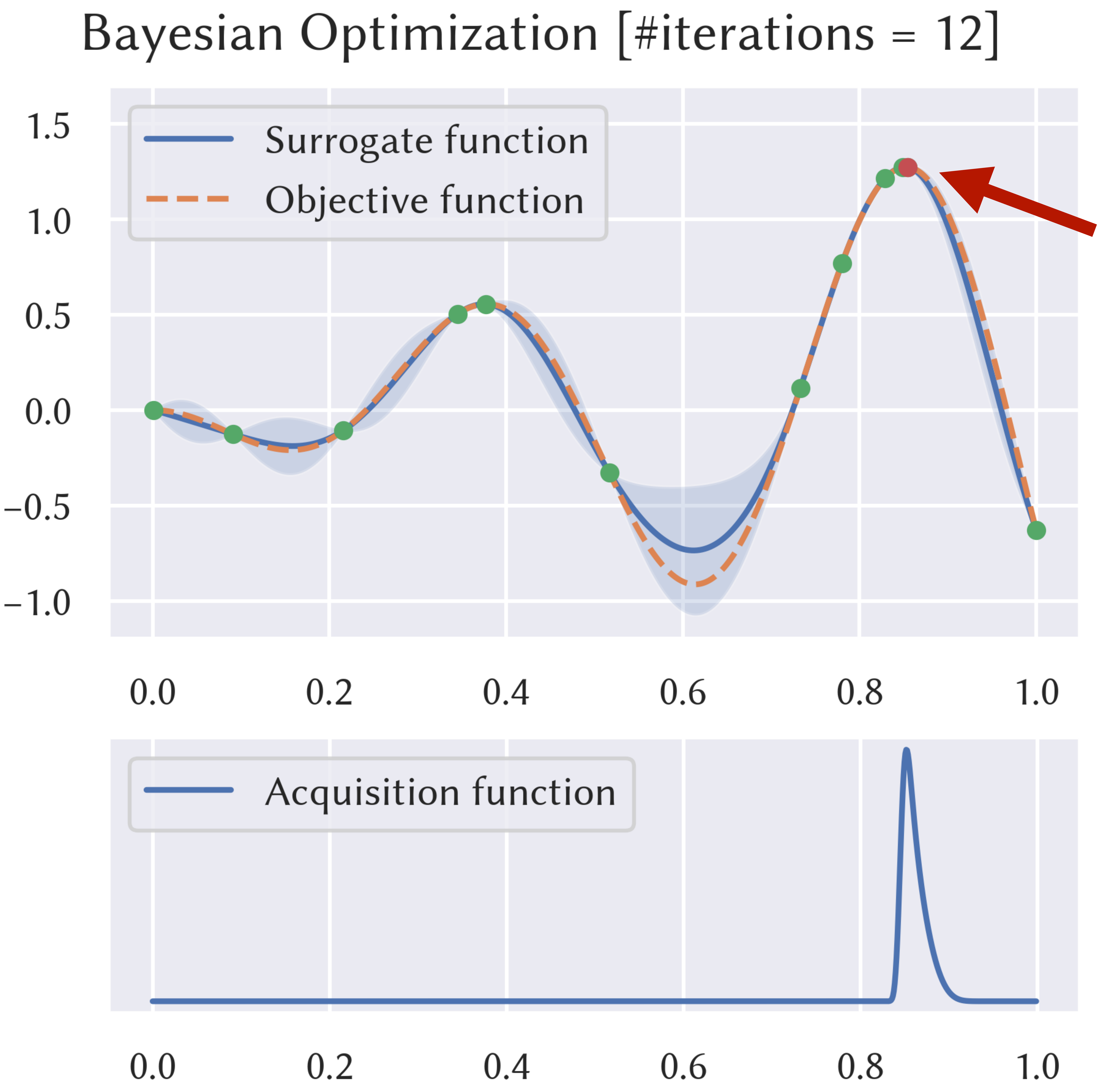


Bayesian Optimization [#iterations = 09]
Surrogate function
Objective function
1.5
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0
Acquisition function
0.0
0.2
0.4
0.6
0.8
1.0



Bayesian Optimization [#iterations = 10]
Surrogate function
Objective function
1.5
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0
Acquisition function
0.0
0.2
0.4
0.6
0.8
1.0


Bayesian Optimization [#iterations = 11]
Surrogate function
Objective function
1.5
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0
Acquisition function
0.0
0.2
0.4
0.6
0.8
1.0



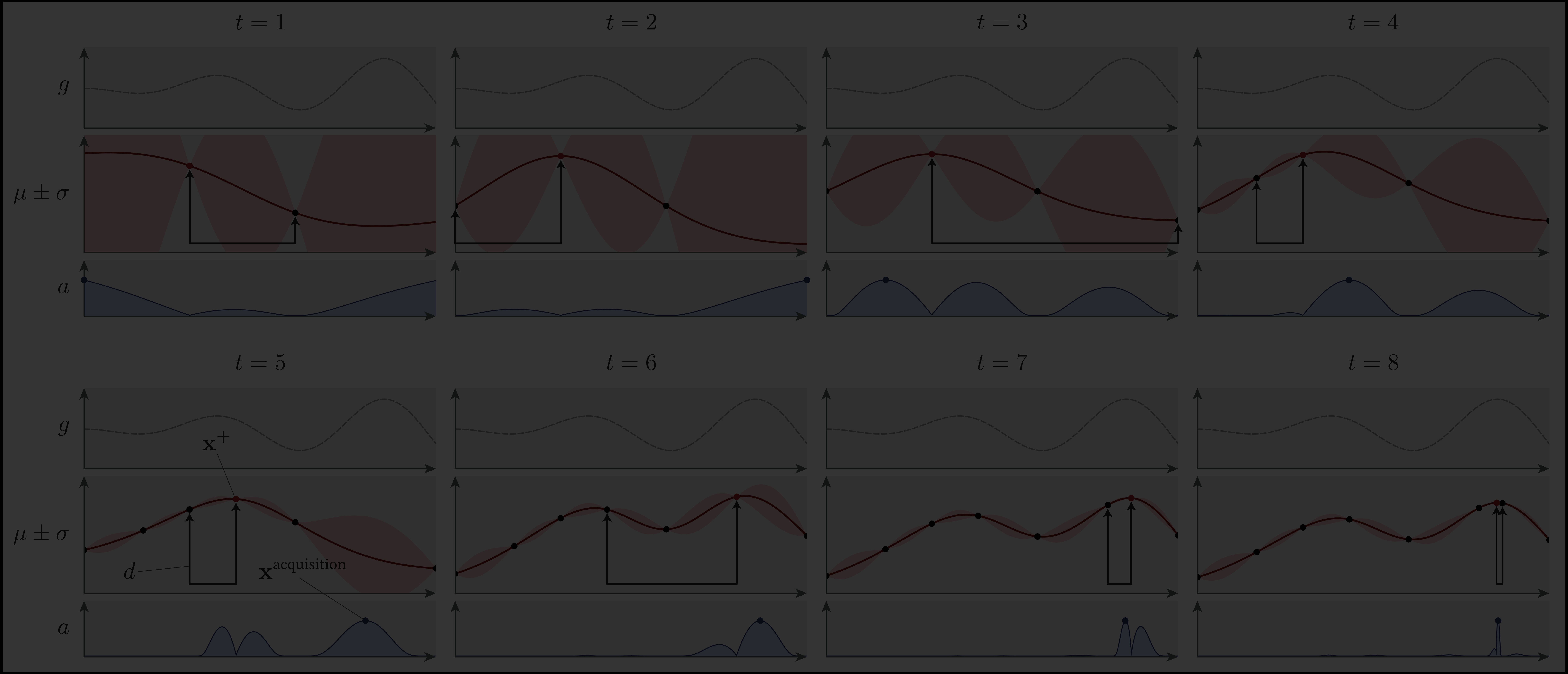
We could find a very nice solution with only 12 iterations in this case

Preferential Bayesian Optimization (PBO)

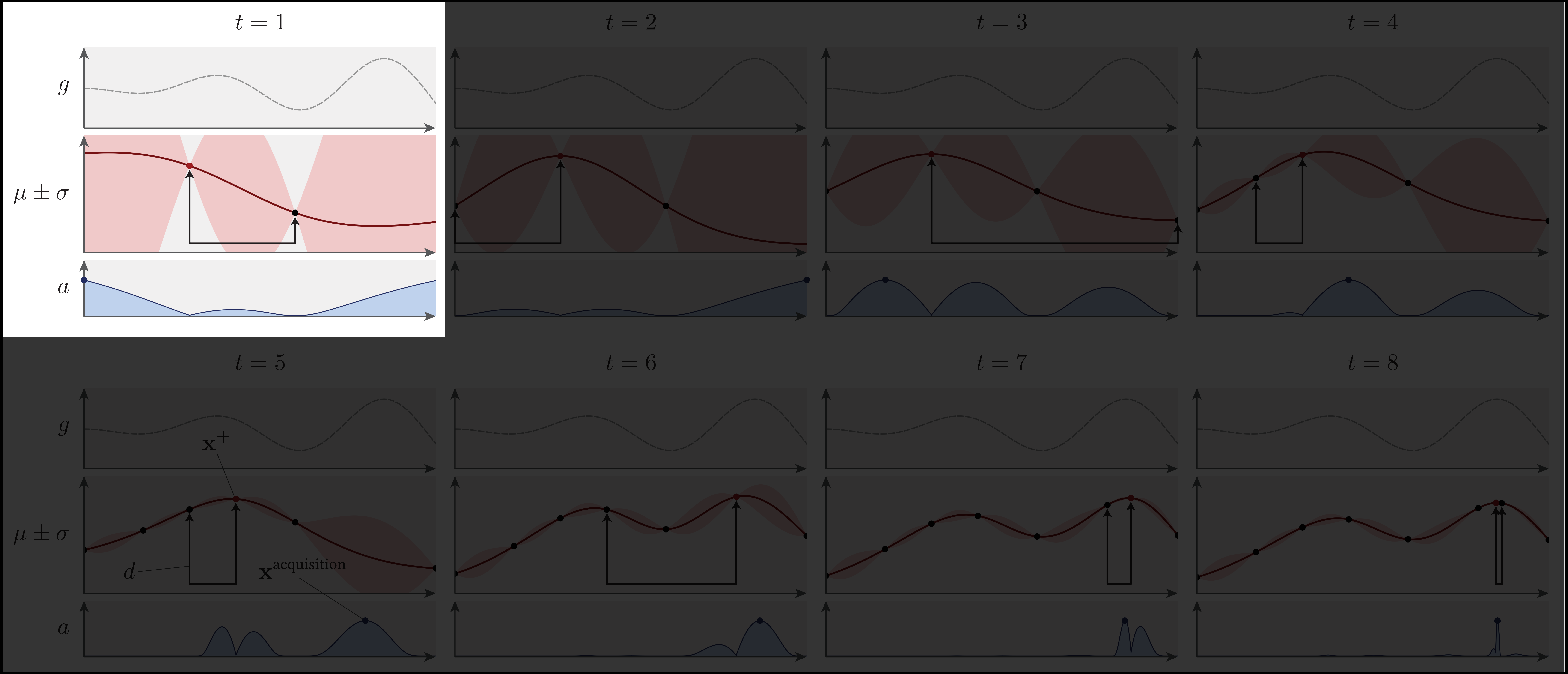
Preferential Bayesian Optimization (PBO)

- PBO is an extension of BO, which **runs with relative assessment (preferential feedback)**, rather than absolute assessment of function values
- PBO can find optimal solutions with **only a small number of preferential feedbacks**
  - Recall: human is expensive-to-query

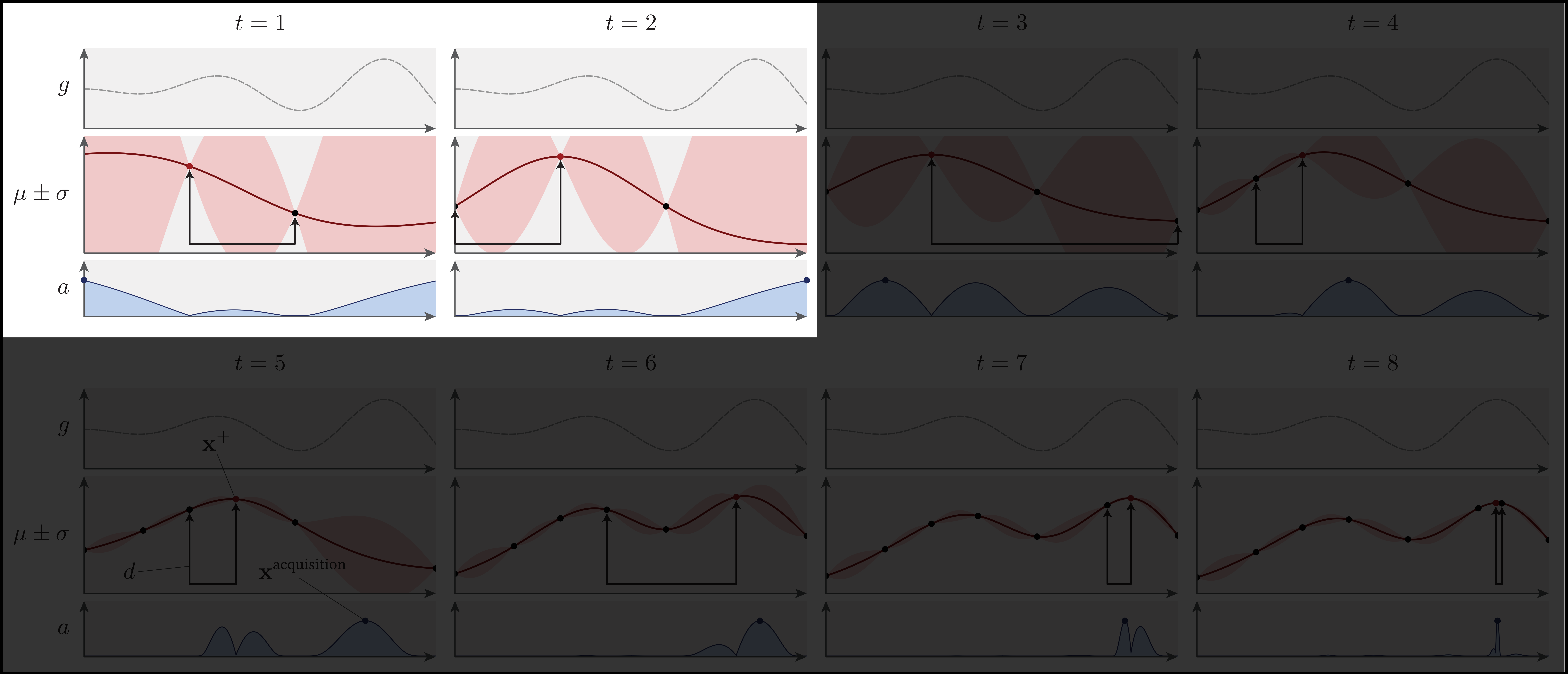
Query: **pairwise comparison** between the current best point and the maximizer of the acquisition function [Brochu+, NIPS 2007]



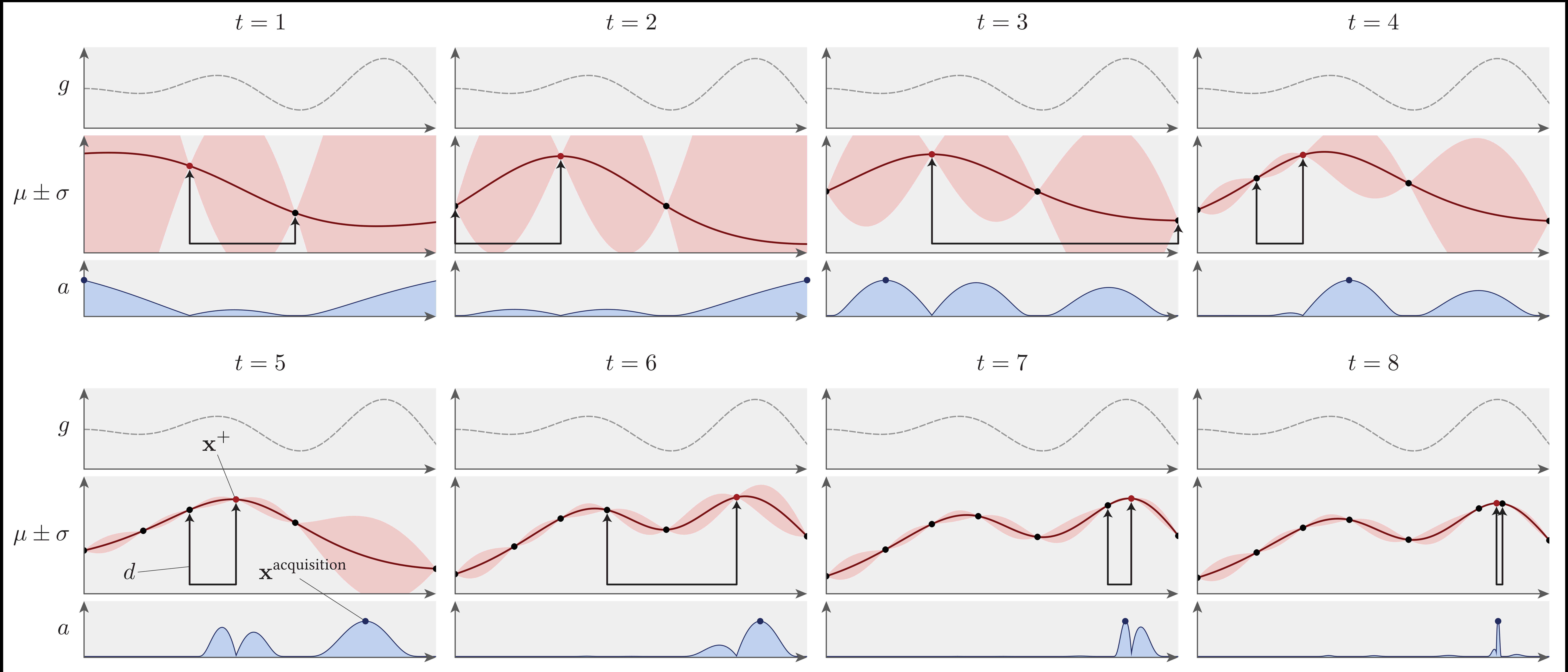
Query: **pairwise comparison** between the current best point and the maximizer of the acquisition function [Brochu+, NIPS 2007]

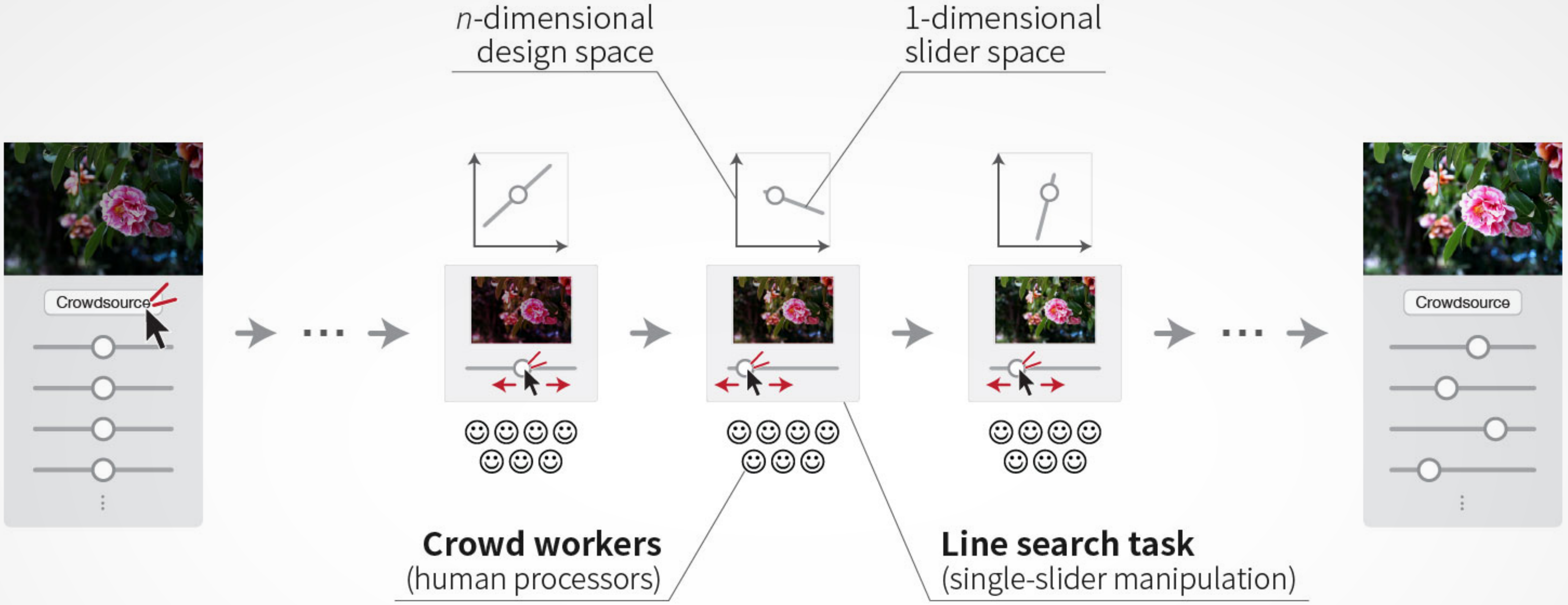


Query: **pairwise comparison** between the current best point and the maximizer of the acquisition function [Brochu+, NIPS 2007]



Query: pairwise comparison between the current best point and the maximizer of the acquisition function [Brochu+, NIPS 2007]



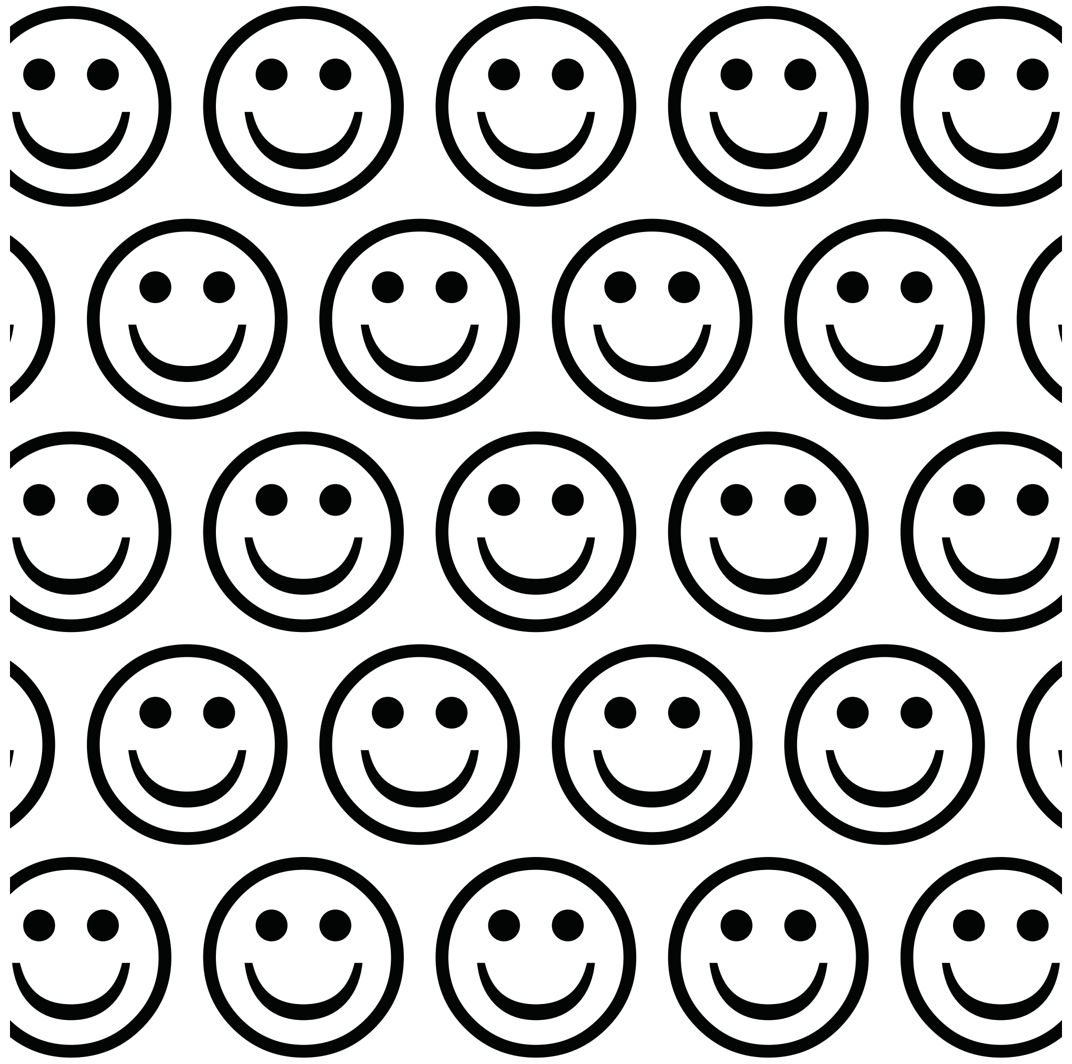


Sequential Line Search for Efficient Visual Design Optimization by Crowds

Y. Koyama, I. Sato, D. Sakamoto, and T. Igarashi

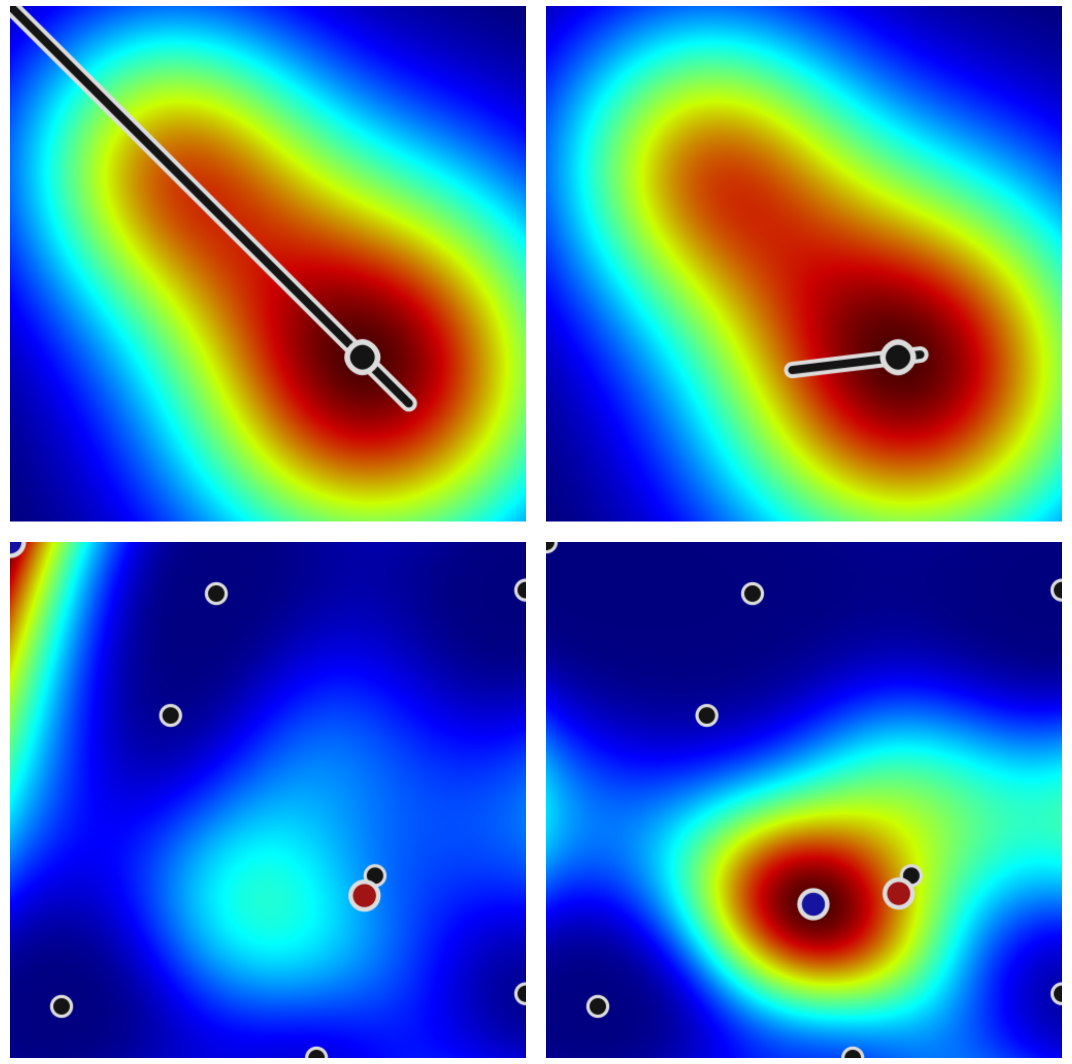
ACM Trans. Graph. (a.k.a. SIGGRAPH 2017)

Contributions



Concept: **Crowd-Powered Design Optimizer**

Solve optimization problems using human computation

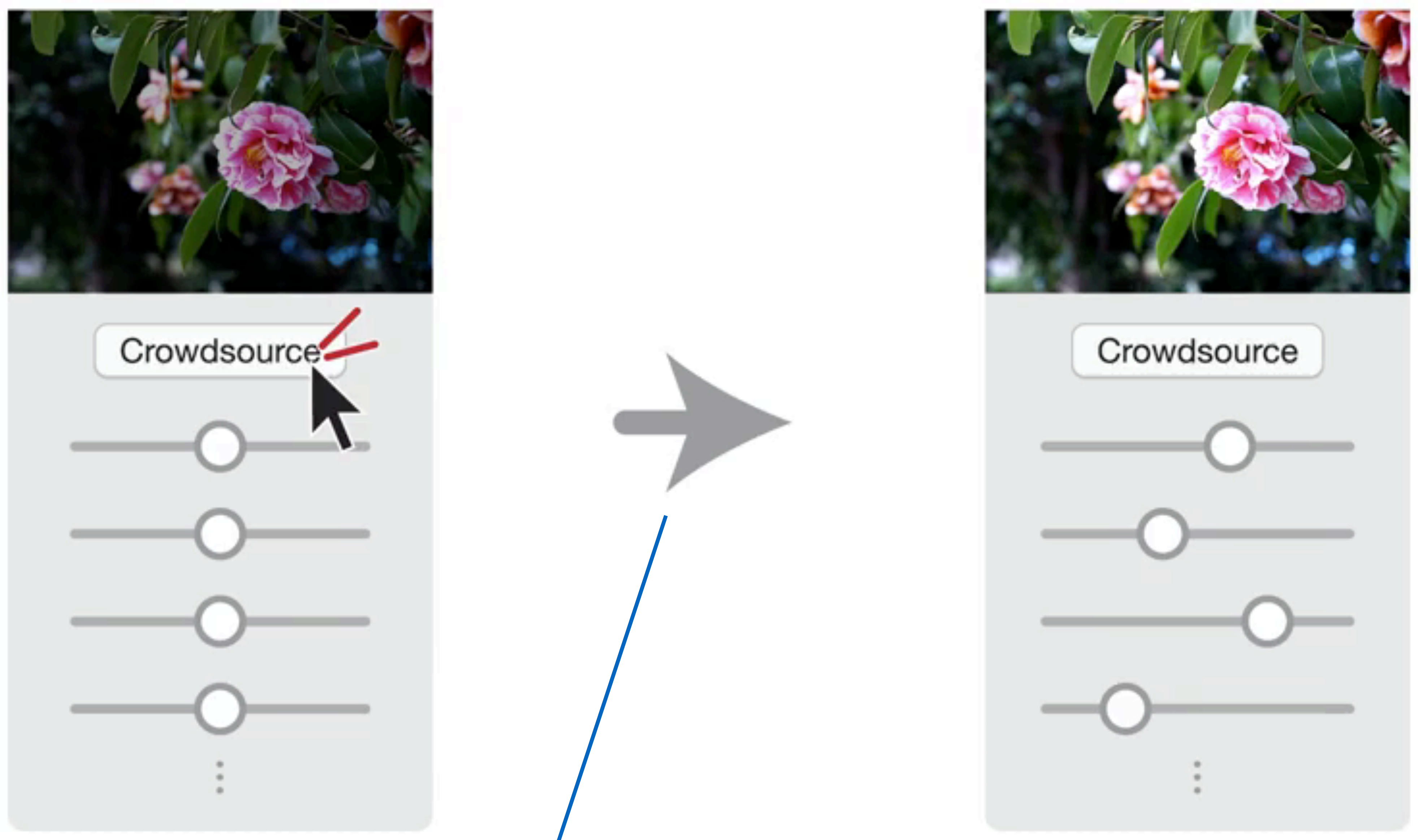


Technique: **Sequential Line Search**

Propose a new extension of **preferential Bayesian optimization (PBO)** that is even more efficient

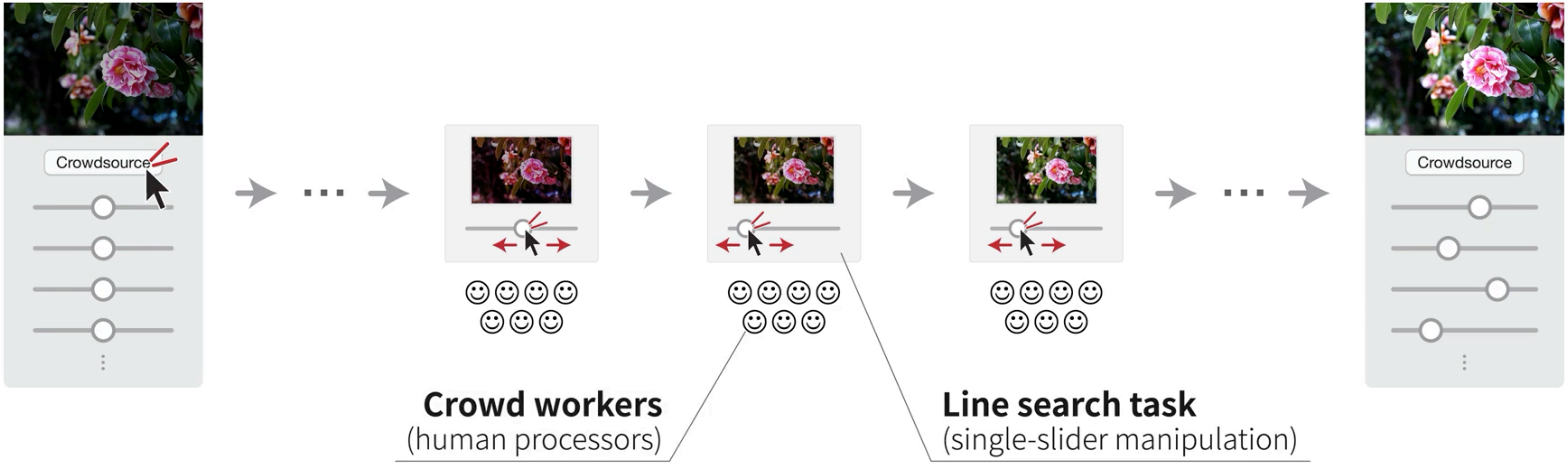
“Crowdsource” button in the tool

“People’s Choice” optimal slider values



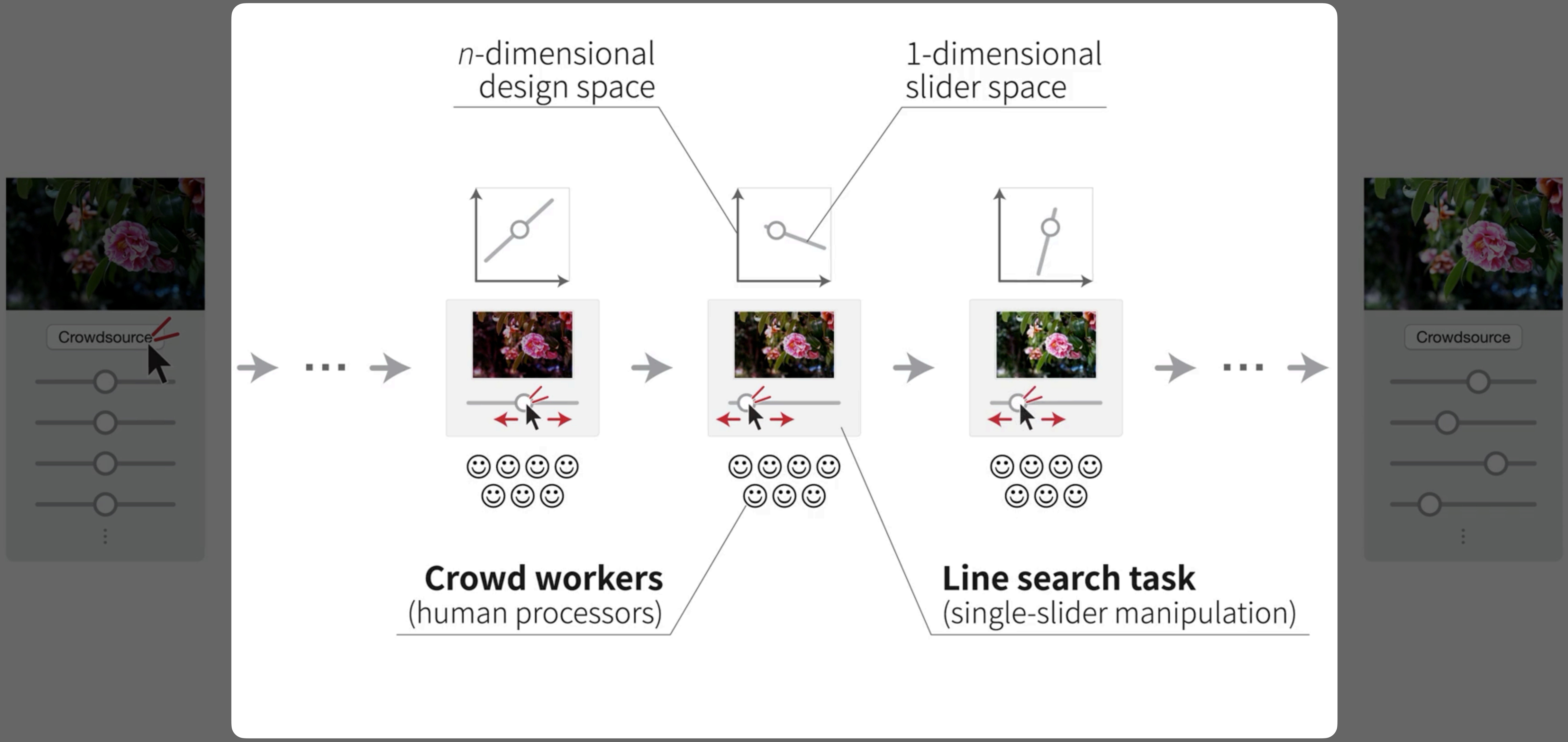
Crowd-in-the-loop optimization

[Koyama+, SIGGRAPH 2017] Yuki Koyama, Issei Sato, Daisuke Sakamoto, and Takeo Igarashi. 2017. Sequential Line Search for Efficient Visual Design Optimization by Crowds. ACM Trans. Graph. 36, 4, pp.48:1–48:11 (2017). https://doi.org/10.1145/3072959.3073598



[Koyama+, SIGGRAPH 2017] Yuki Koyama, Issei Sato, Daisuke Sakamoto, and Takeo Igarashi. 2017. Sequential Line Search for Efficient Visual Design Optimization by Crowds. ACM Trans. Graph. 36, 4, pp.48:1–48:11 (2017). https://doi.org/10.1145/3072959.3073598

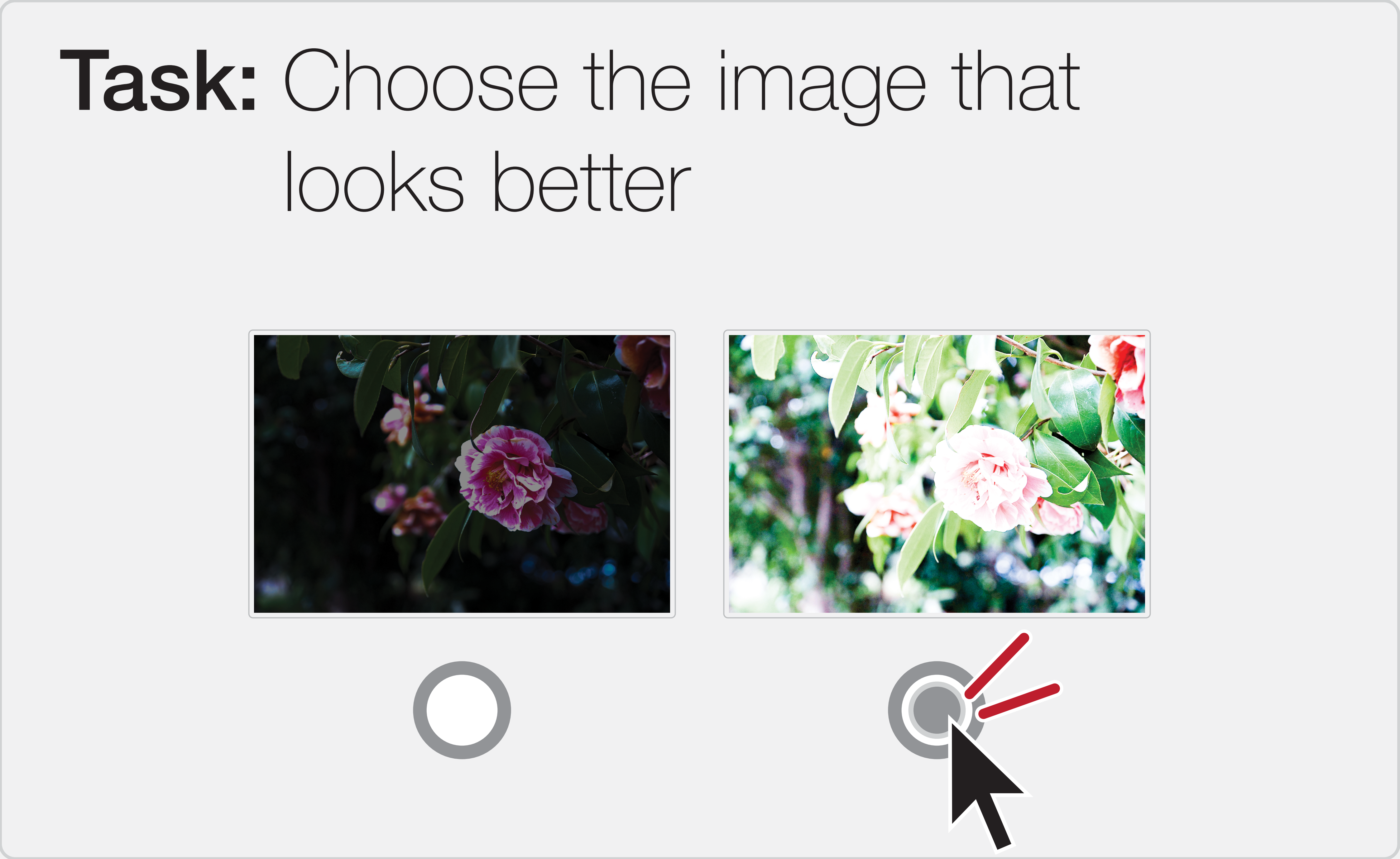
Sequential line search (a new PBO extension)



[Koyama+, SIGGRAPH 2017] Yuki Koyama, Issei Sato, Daisuke Sakamoto, and Takeo Igarashi. 2017. Sequential Line Search for Efficient Visual Design Optimization by Crowds. ACM Trans. Graph. 36, 4, pp.48:1–48:11 (2017). https://doi.org/10.1145/3072959.3073598

An Advanced "Relative Assessment" Microtask Design

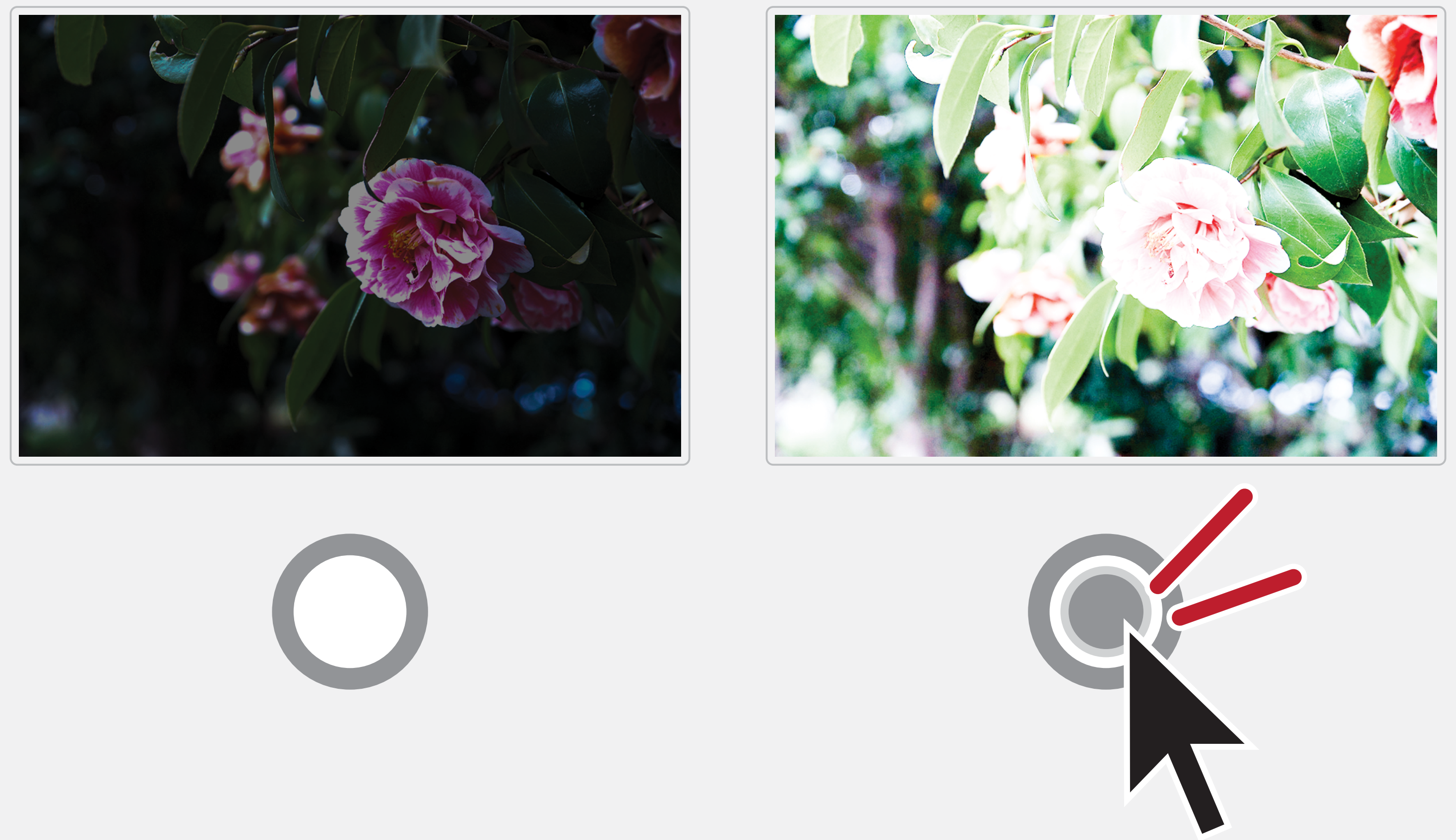
An Advanced "Relative Assessment" Microtask Design



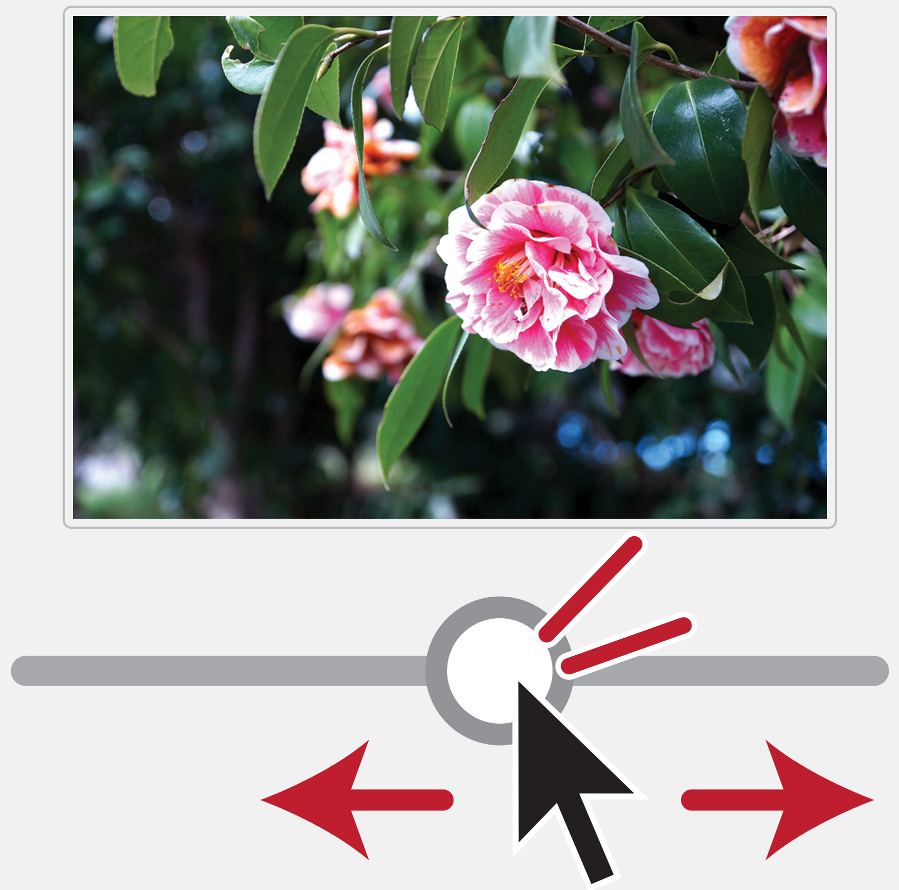
Basic: Pairwise comparison
(e.g., [Brochu+, NIPS 2007])

An Advanced "Relative Assessment" Microtask Design

Task: Choose the image that looks better



Task: Adjust the slider so that the image looks the best

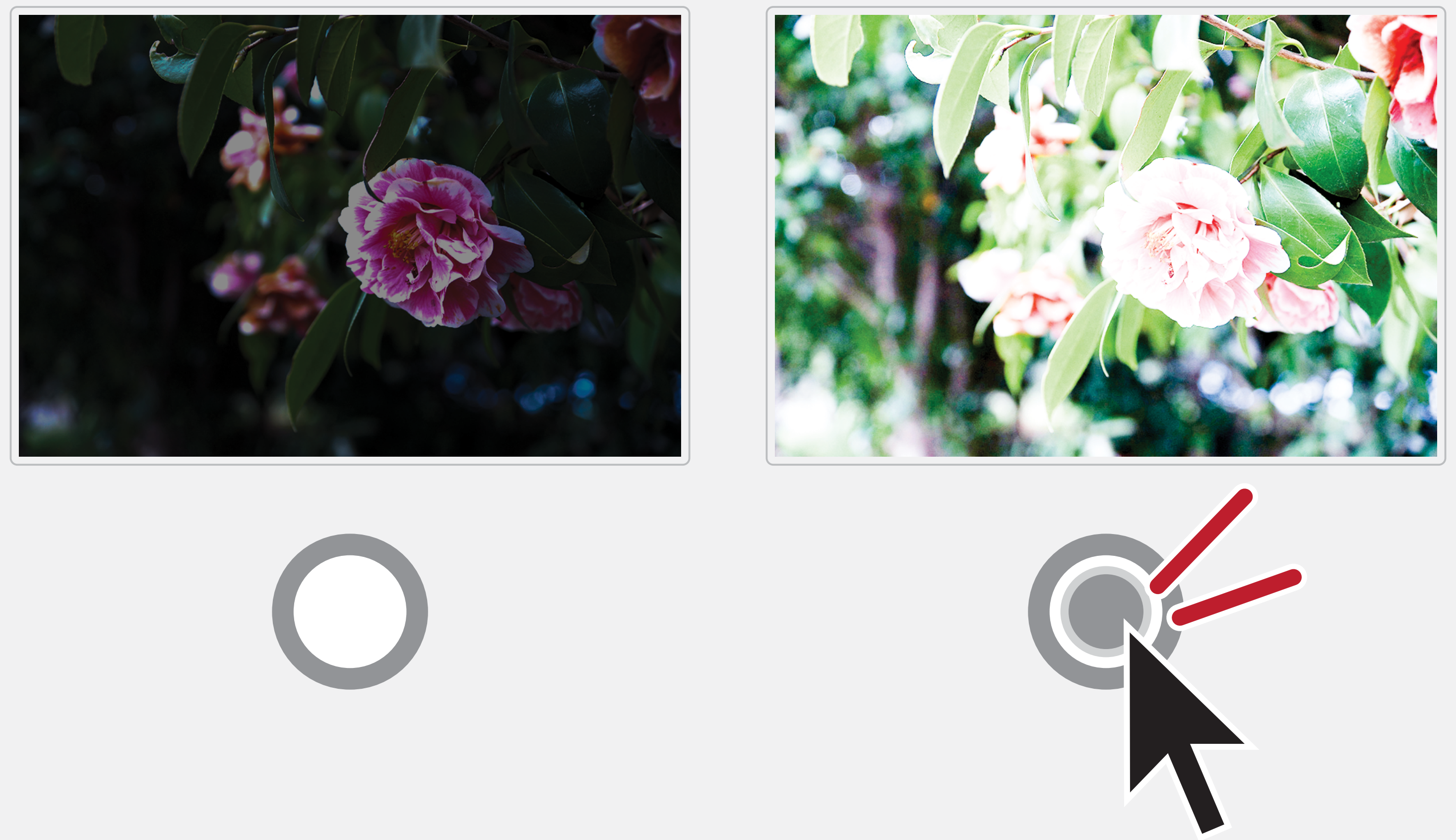


Basic: Pairwise comparison
(e.g., [Brochu+, NIPS 2007])

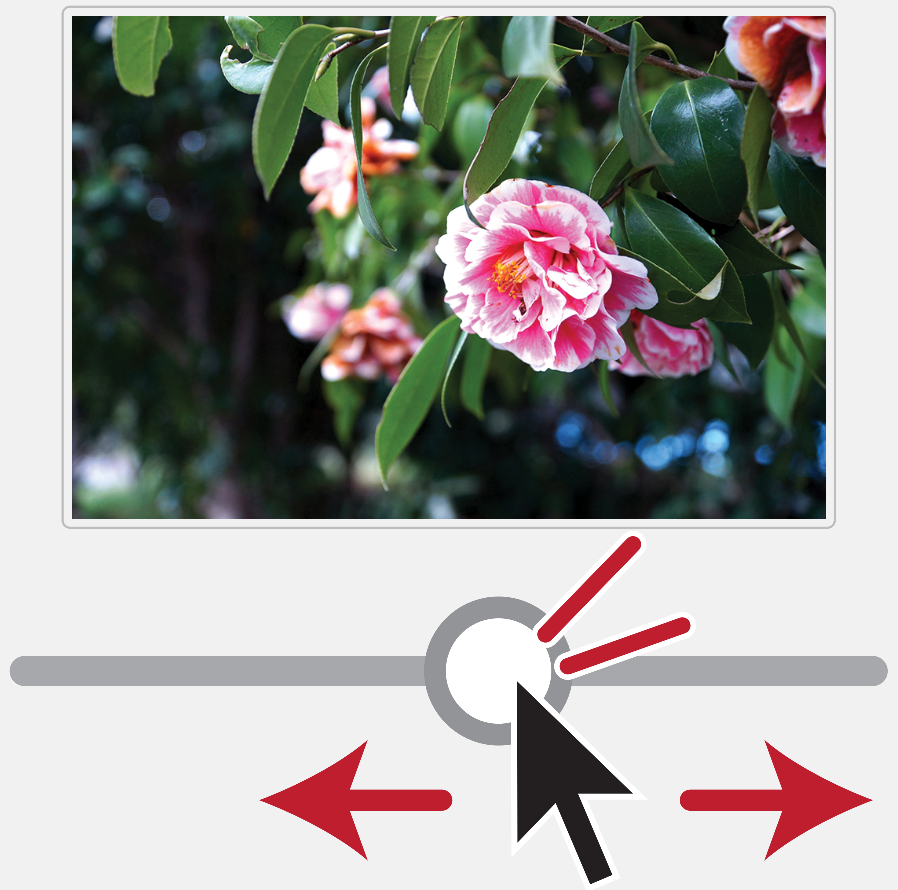
Advanced: Single-slider manipulation
(provides much richer information)

An Advanced "Relative Assessment" Microtask Design

Task: Choose the image that looks better



Task: Adjust the slider so that the image looks the best

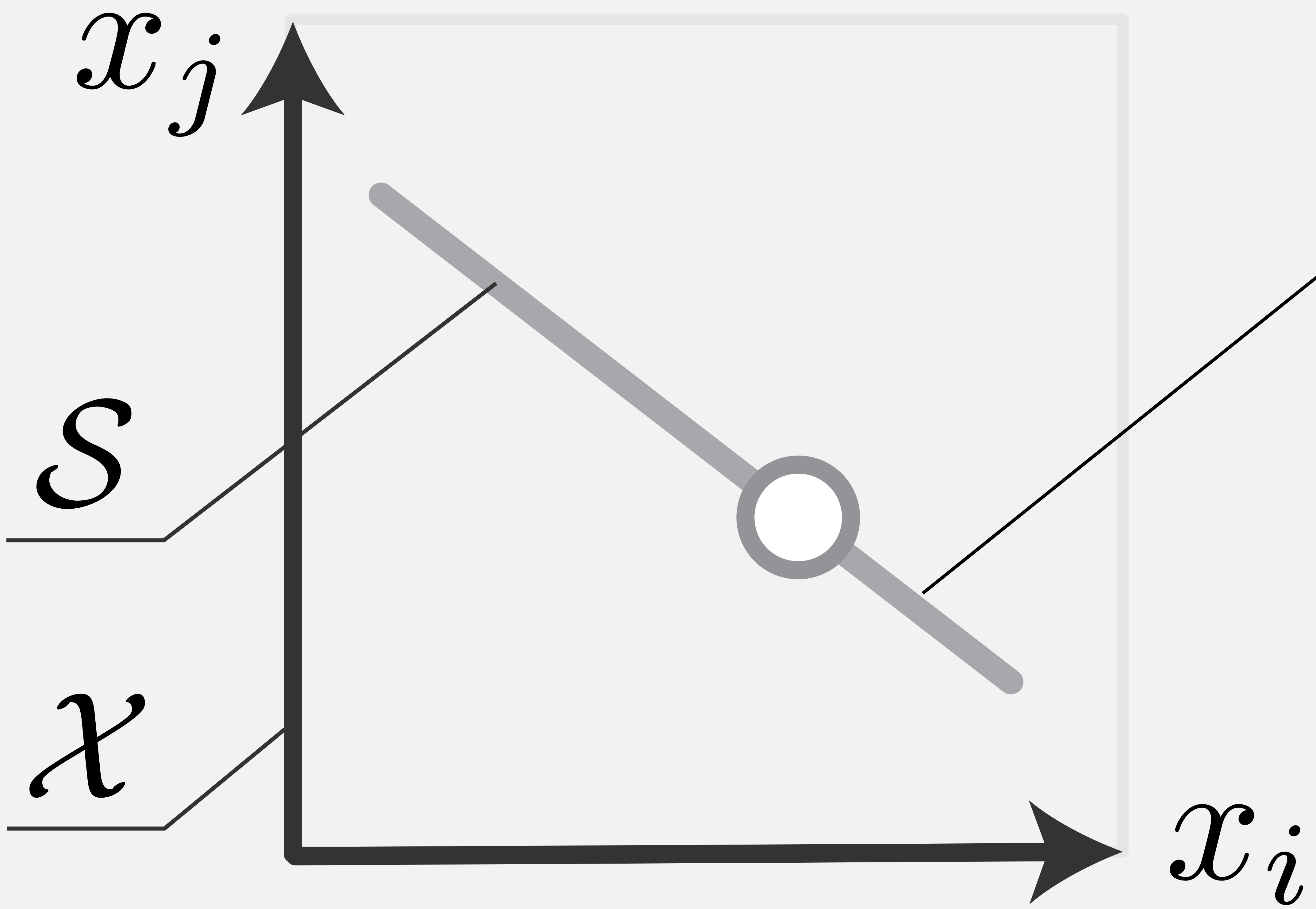


Basic: **Pairwise comparison**
(e.g., [Brochu+, NIPS 2007])

Advanced: **Single-slider manipulation**
(provides much richer information)

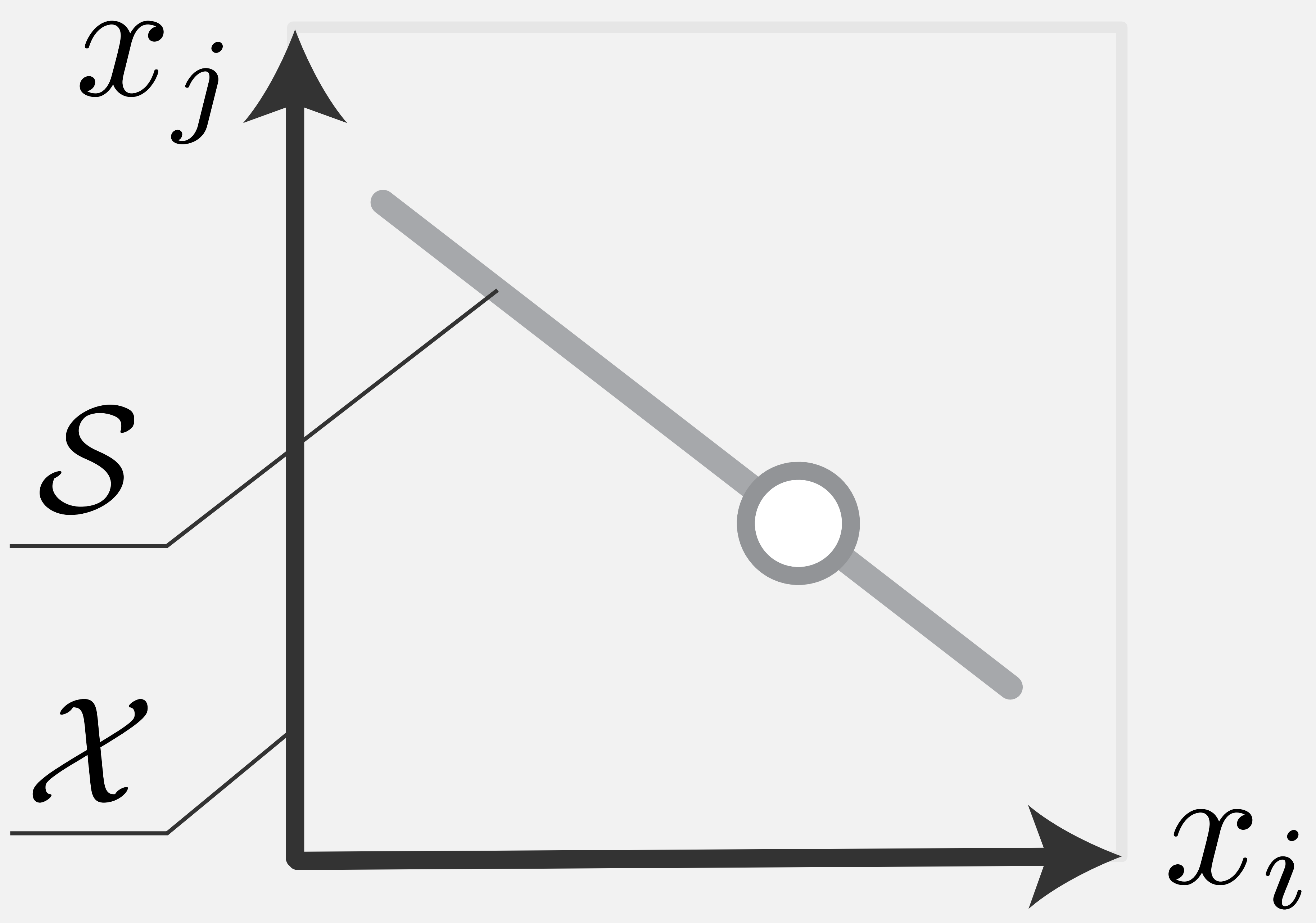
➡ **Faster convergence**

How to Define Slider Spaces

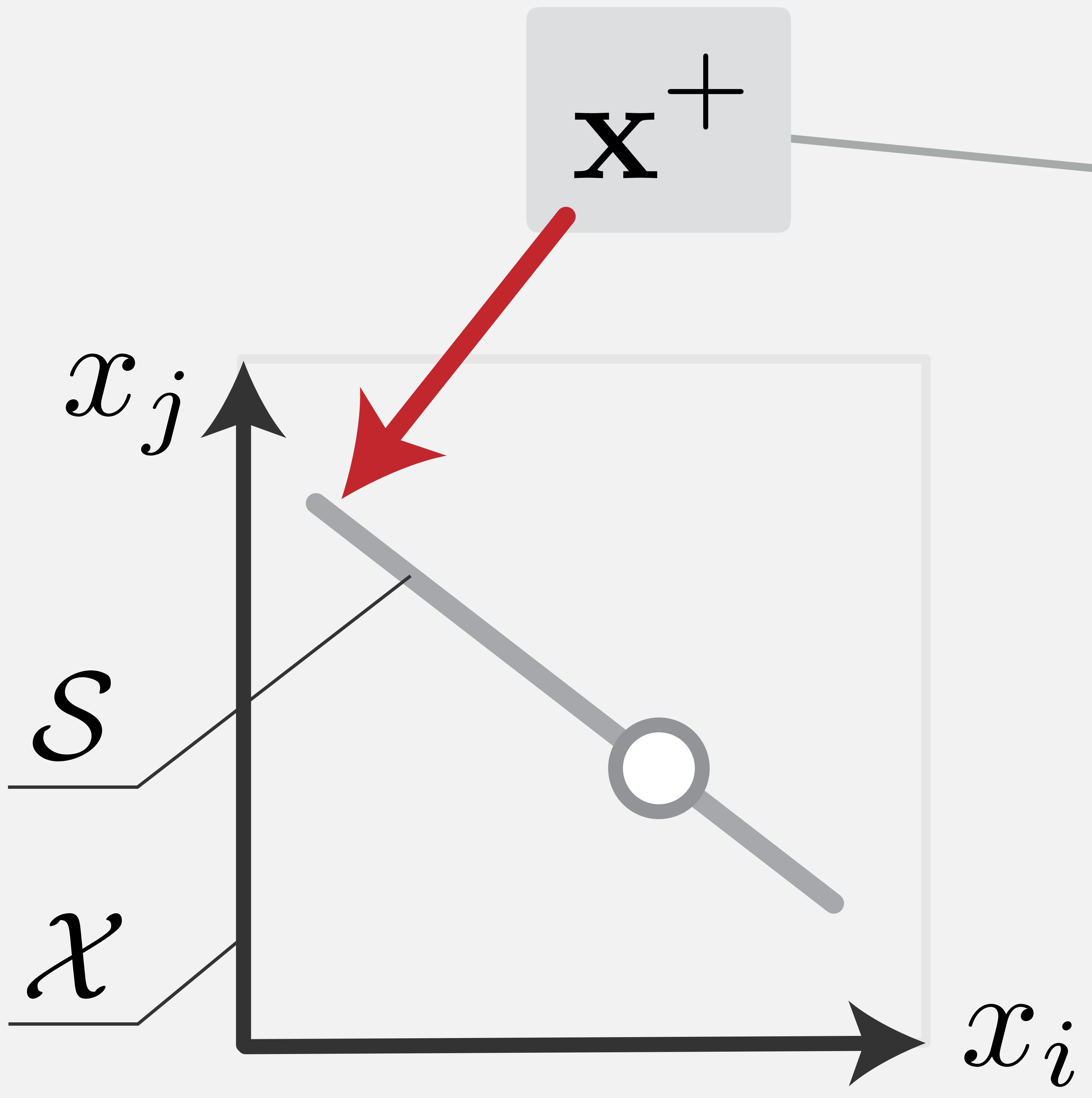


Slider space S **:**
1D subspace
mapped to a slider

How to Define Slider Spaces



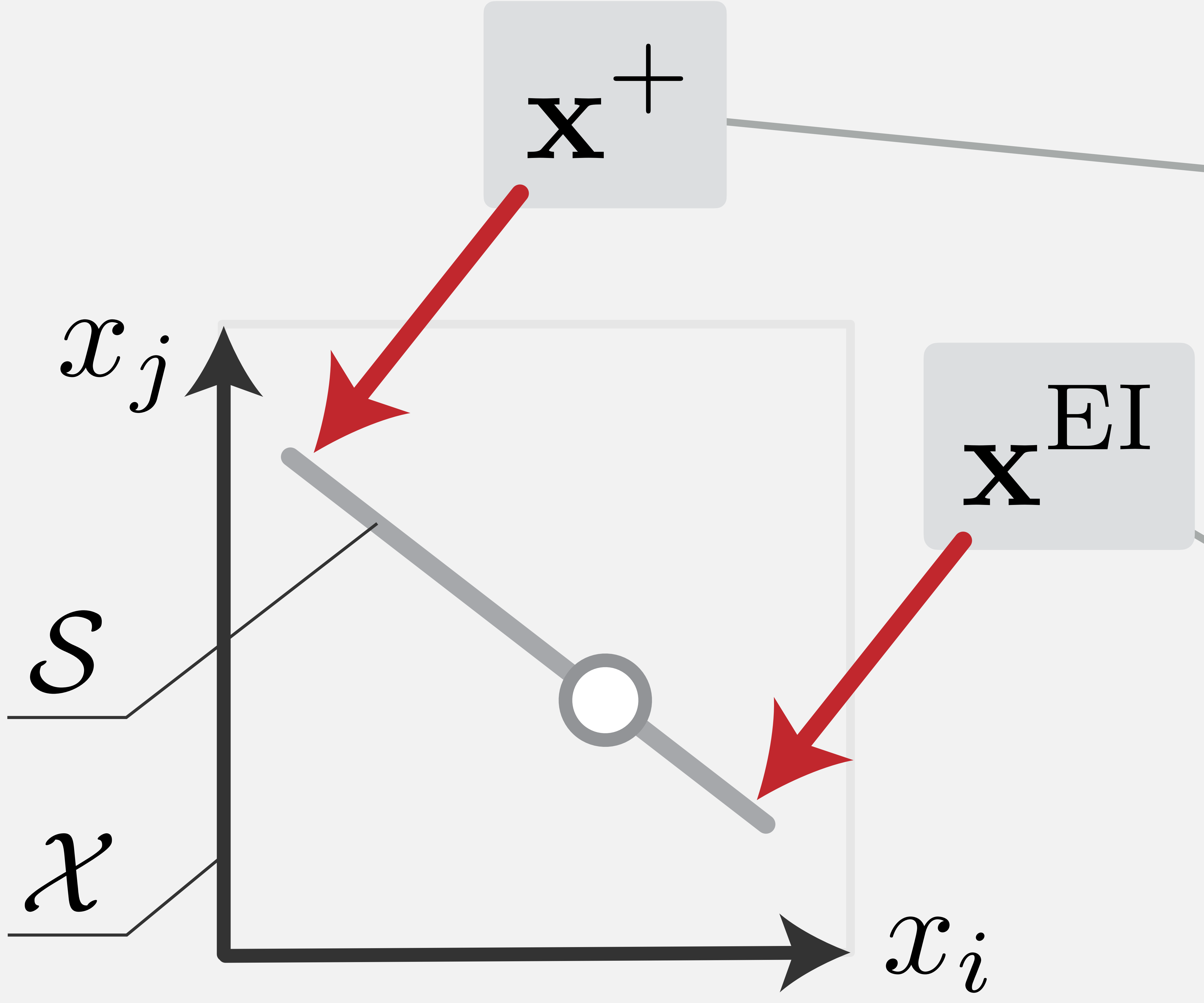
How to Define Slider Spaces



$$\mathbf{x}^+ = \arg\max_{\mathbf{x}\in\{\mathbf{x}_i\}} \mu(\mathbf{x})$$

The best point among the already visited points

How to Define Slider Spaces



$$\mathbf{x}^{+} = \mathop{\arg\max}_{\mathbf{x}\in\{\mathbf{x}_i\}} \mu(\mathbf{x})$$

The best point among the already visited points

$$\mathbf{x}^{\mathrm{EI}} = \mathop{\arg\max}_{\mathbf{x}\in\mathcal{X}} \mathrm{EI}(\mathbf{x})$$

The point that maximizes the "**expected improvement**" (EI)

Applications #1

Photo Color Enhancement (6D)

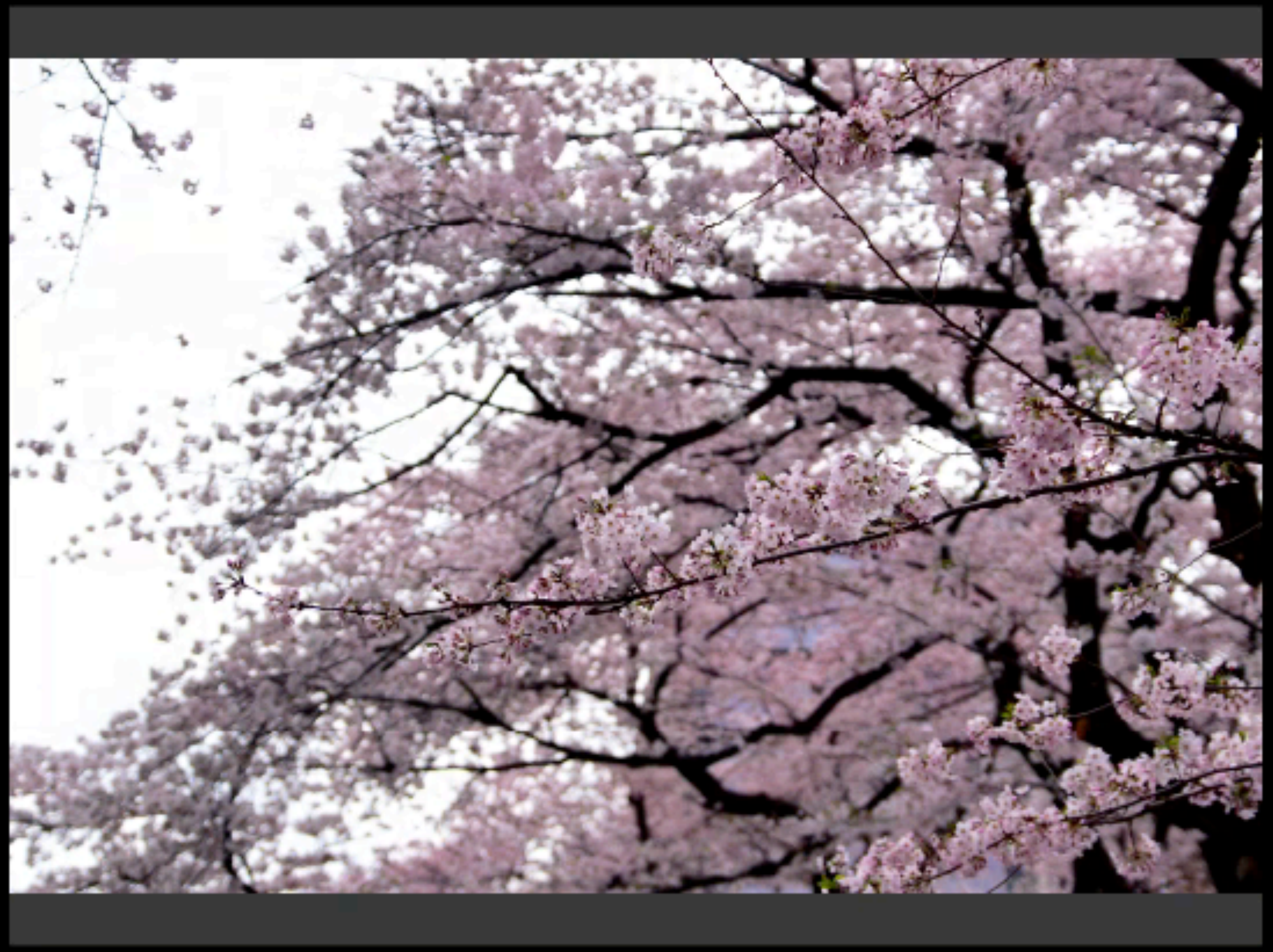
Original Photographs

For each photo, it runs **15 iterations**, cost **5.25 USD** in total, and took **68 min** in average

Results








For each photo, it runs **15 iterations**, cost **5.25 USD** in total, and took **68 min** in average

Replay

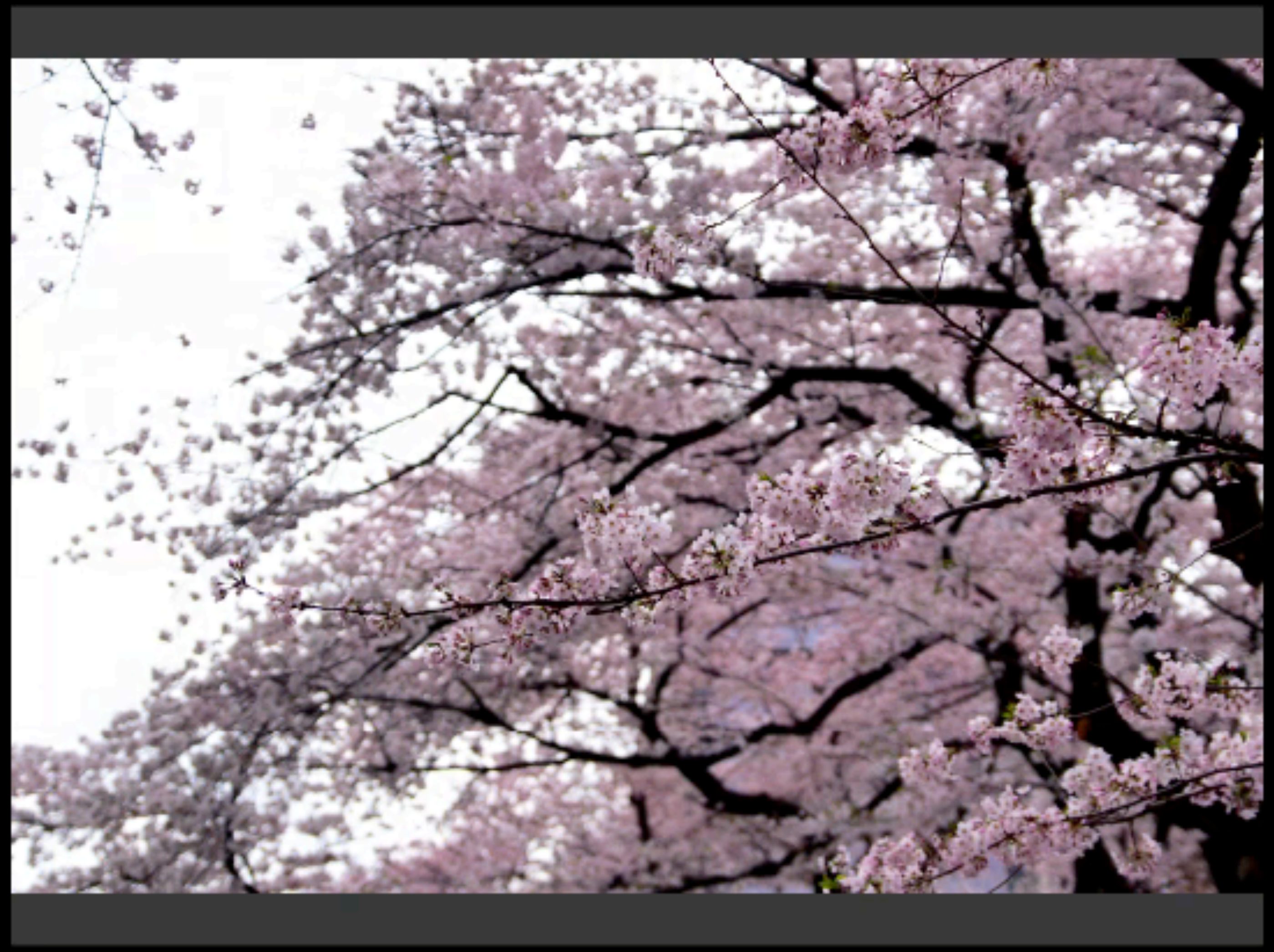
Original Photographs

For each photo, it runs **15 iterations**, cost **5.25 USD** in total, and took **68 min** in average

Results








For each photo, it runs **15 iterations**, cost **5.25 USD** in total, and took **68 min** in average

Evaluation: Crowdsourced Voting

Q. Which one do you like?

Original

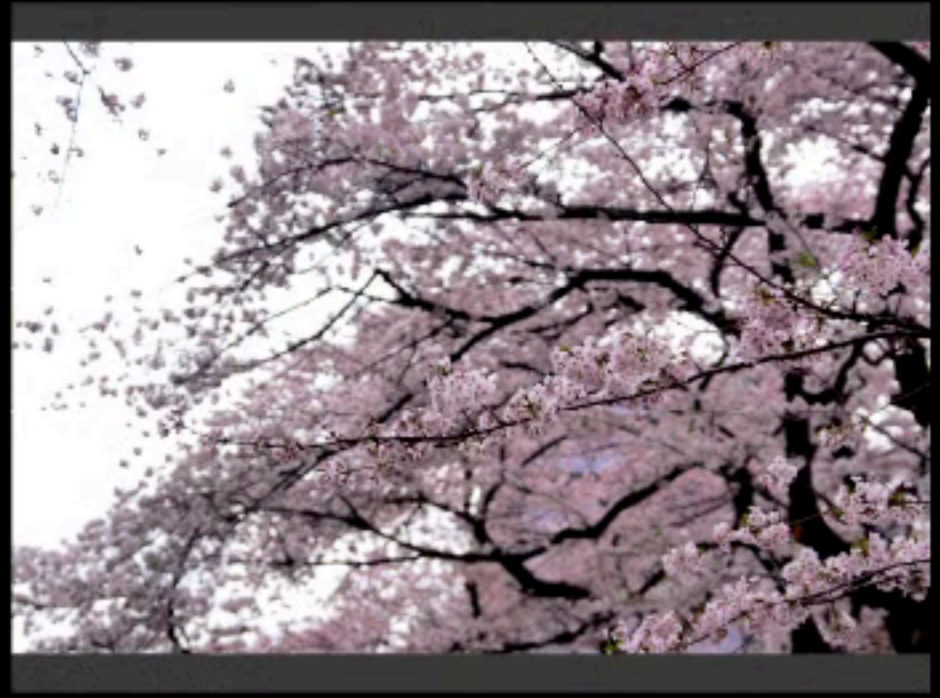
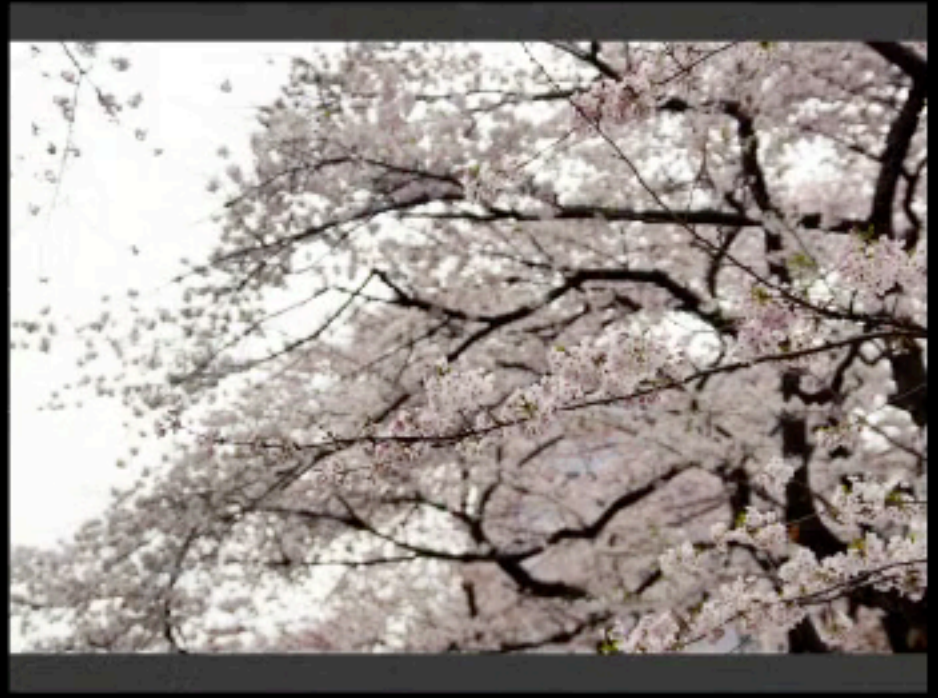
By Crowds

By Photoshop

By Lightroom

	Original	Crowds	Photoshop	Lightroom
				
Preferred by:	2	26	2	3
				
Preferred by:	0	32	0	1

	Original	Crowds	Photoshop	Lightroom
				
Preferred by:	0	29	1	3
				
Preferred by:	1	23	3	6

	Original	Crowds	Photoshop	Lightroom
				
Preferred by:	0	29	3	1
				
Preferred by:	0	31	0	2

Q. Which one do you like?

Original	By Crowds	By Photoshop	By Lightroom
			

➡ **As they are the “people’s choice” enhancements, people preferred them**

Applications #2

Material Appearance (3D / 7D)

Reference

Reference

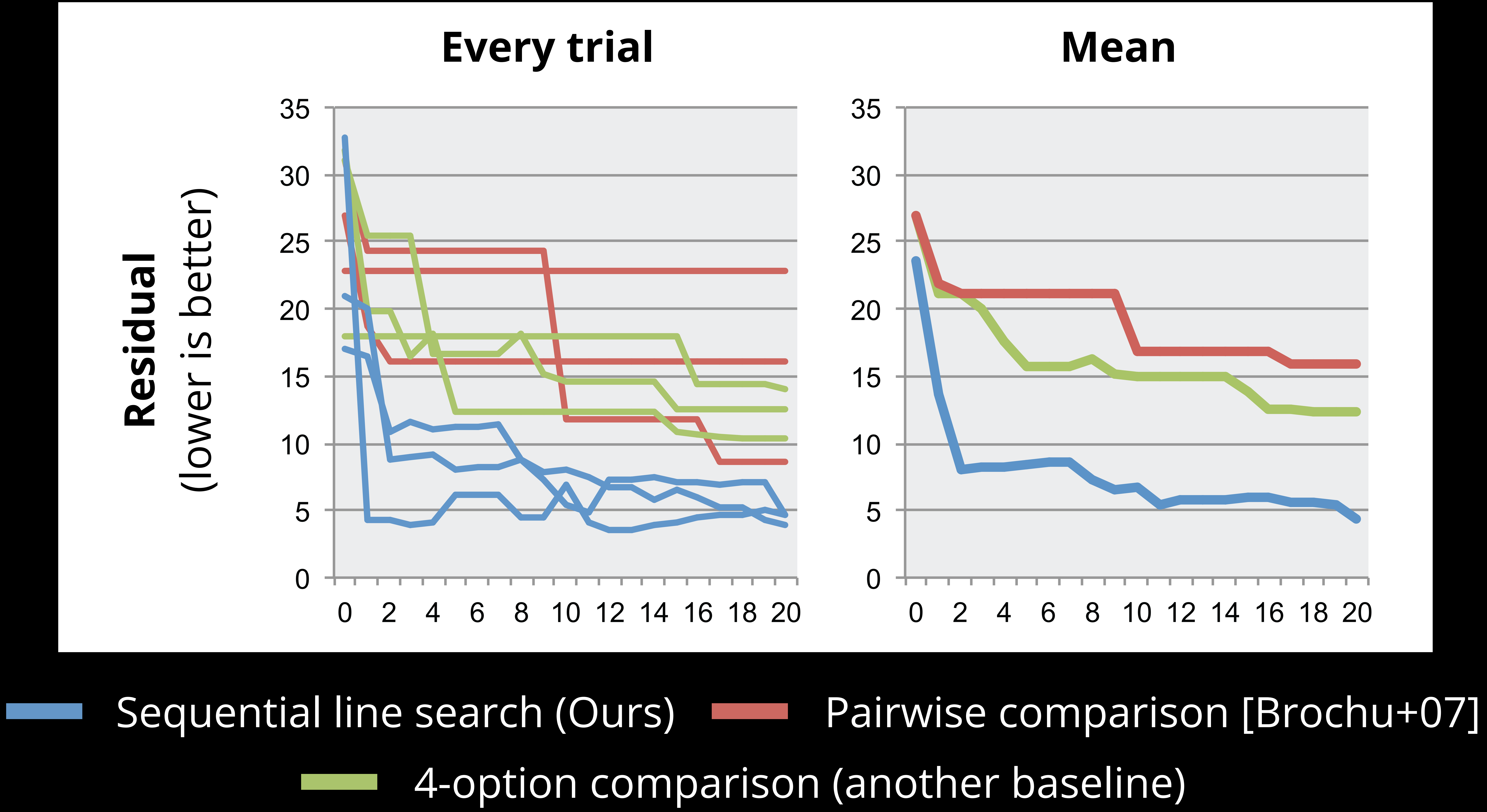
Result (3D)

Result (3D)

Evaluation

Efficiency Comparison

Experiment: Photo Color Enhancement with a Reference



$$\max_{\mathbf{x}} \ \mathrm{Preference}(\mathbf{x})$$

Solve this design problem by a new human-in-the-loop optimization technique called **sequential line search**, which is a new extension of **preferential Bayesian optimization (PBO)**

$$\max_{\mathbf{x}} \ \mathrm{Preference}(\mathbf{x})$$

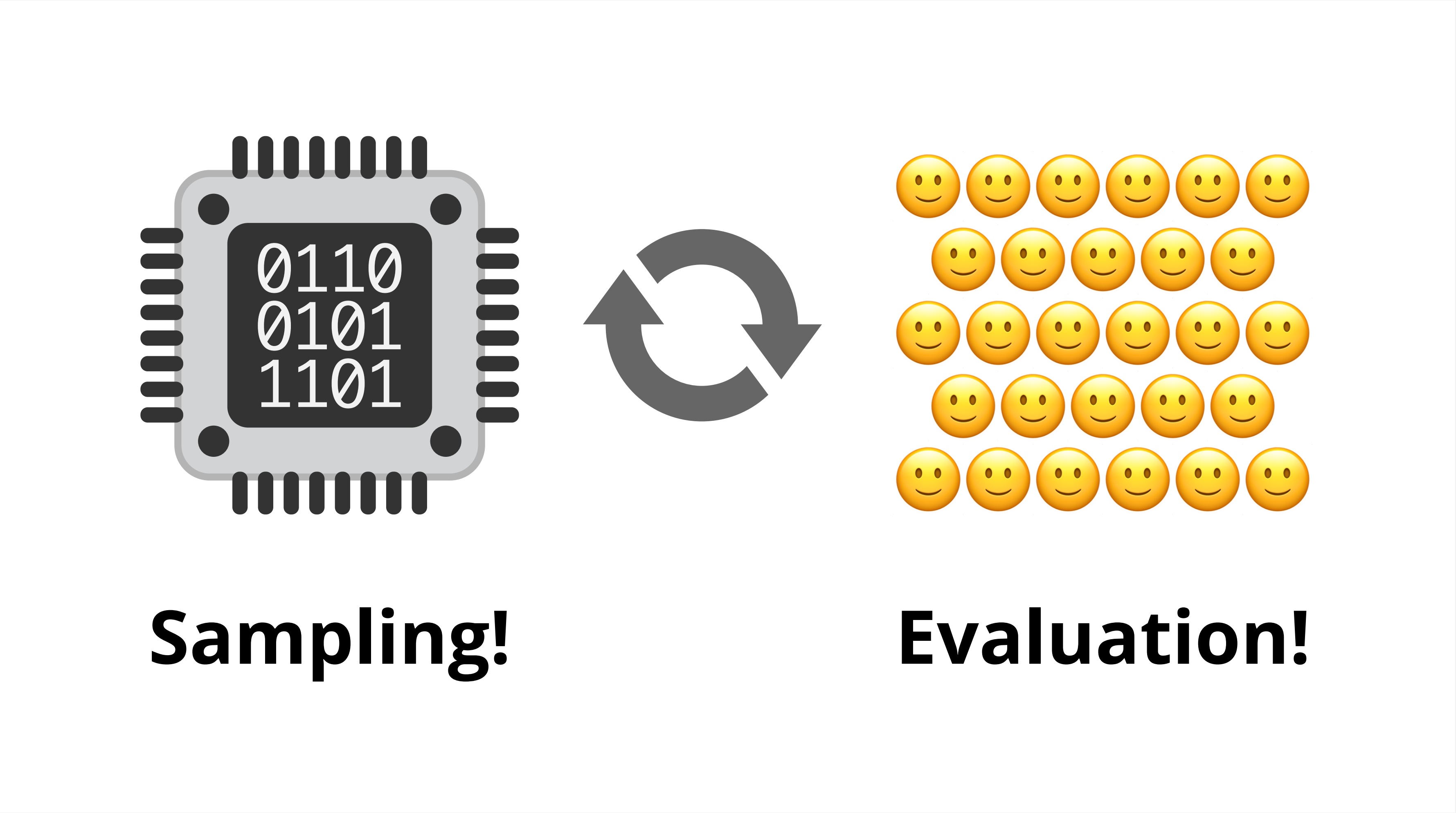
Solve this design problem by a new human-in-the-loop optimization technique called **sequential line search**, which is a new extension of **preferential Bayesian optimization (PBO)**

The “line search” task design was simple enough for crowdsourcing and was useful to **reduce the number of necessary iterations**

Evaluators in Human-in-the-Loop Optimization

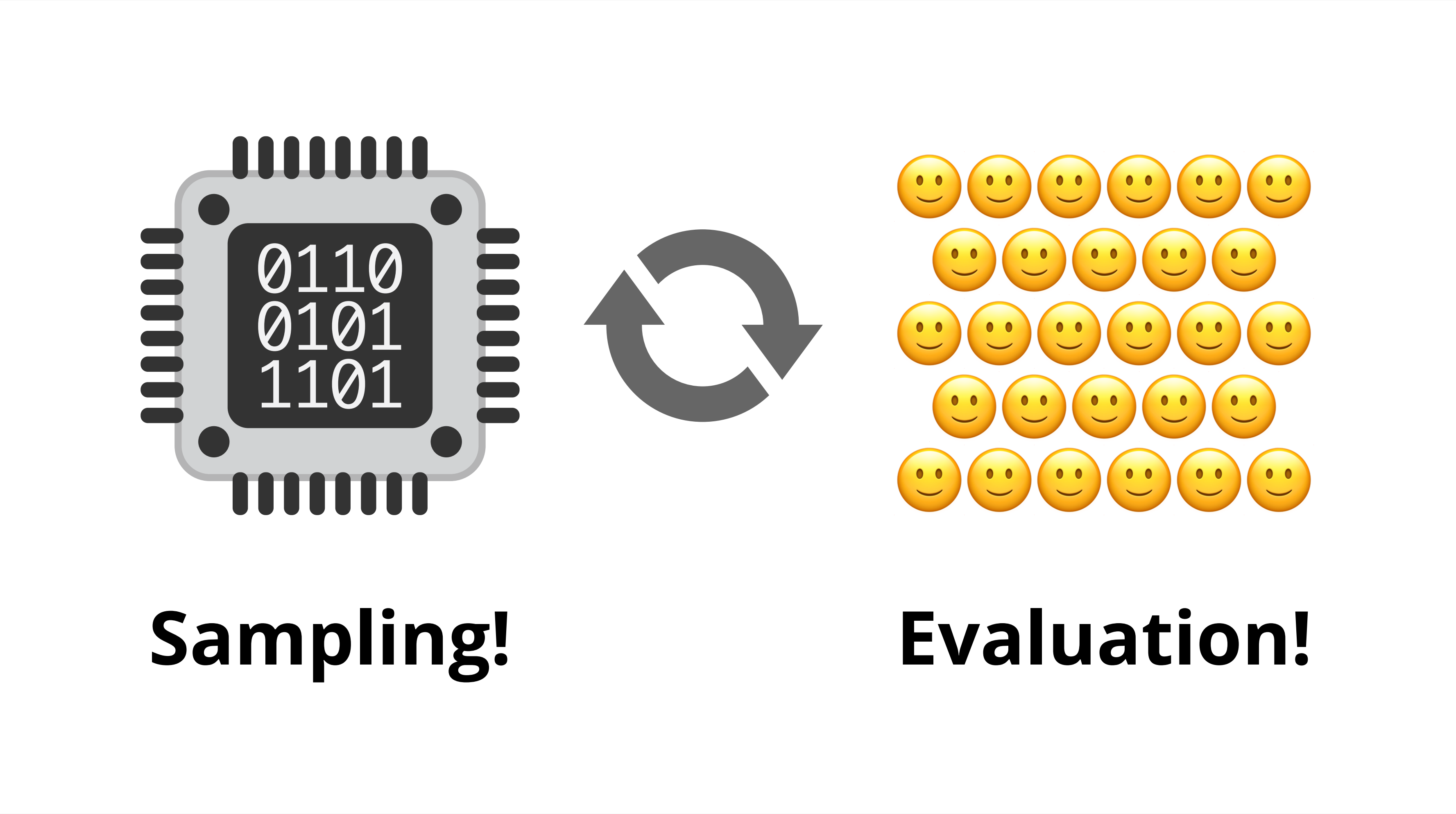
Evaluators

Evaluators

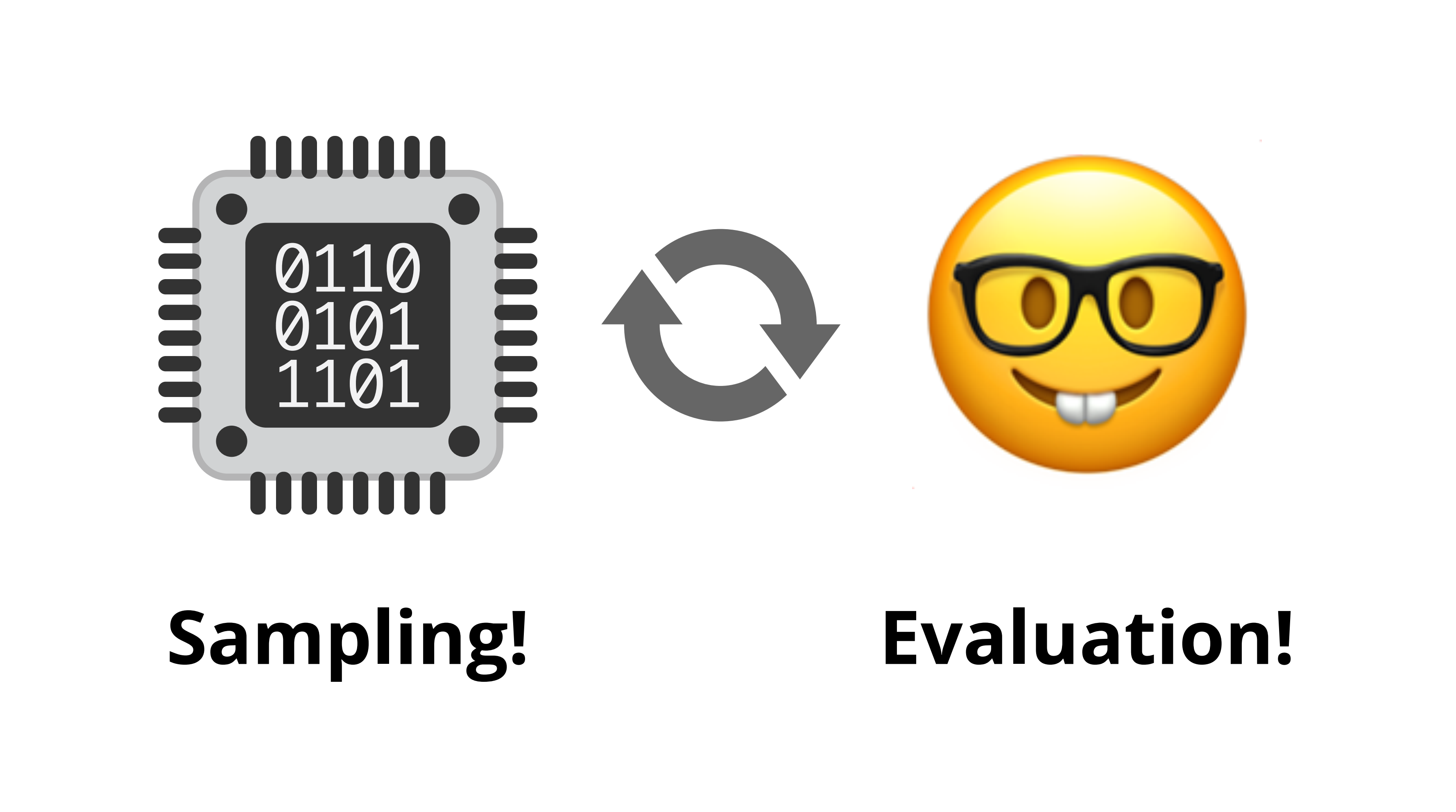


- Crowd-in-the-loop optimization
- Features:
  - General (average) preference
  - Automatic execution
- Task design: **should be simple enough**

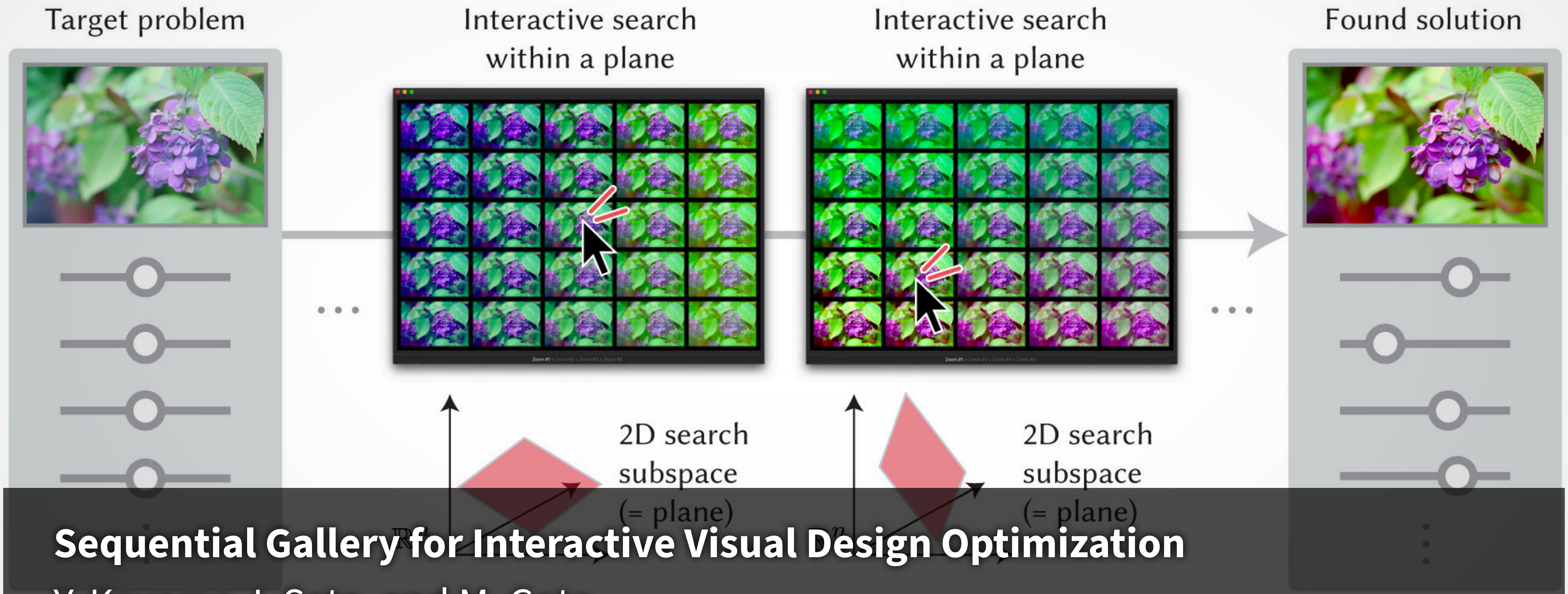
Evaluators



- **Crowd-in-the-loop optimization**
- Features:
  - General (average) preference
  - Automatic execution
- Task design: **should be simple enough**



- **User-in-the-loop optimization**
- Features:
  - Personal preference
  - Interactive execution
- Task design: **can be more complex**



Sequential Gallery for Interactive Visual Design Optimization

Y. Koyama, I. Sato, and M. Goto

ACM Trans. Graph. (a.k.a. SIGGRAPH 2020)

Overview

Proposed System: Sequential Gallery

Target:
n design parameters
(e.g., photo enhance)

Output:
An optimal
parameter set

Target:
n design parameters
(e.g., photo enhance)

Sequential Gallery:
An interactive optimization framework
where the user sequentially performs
2D search subtasks via a grid interface

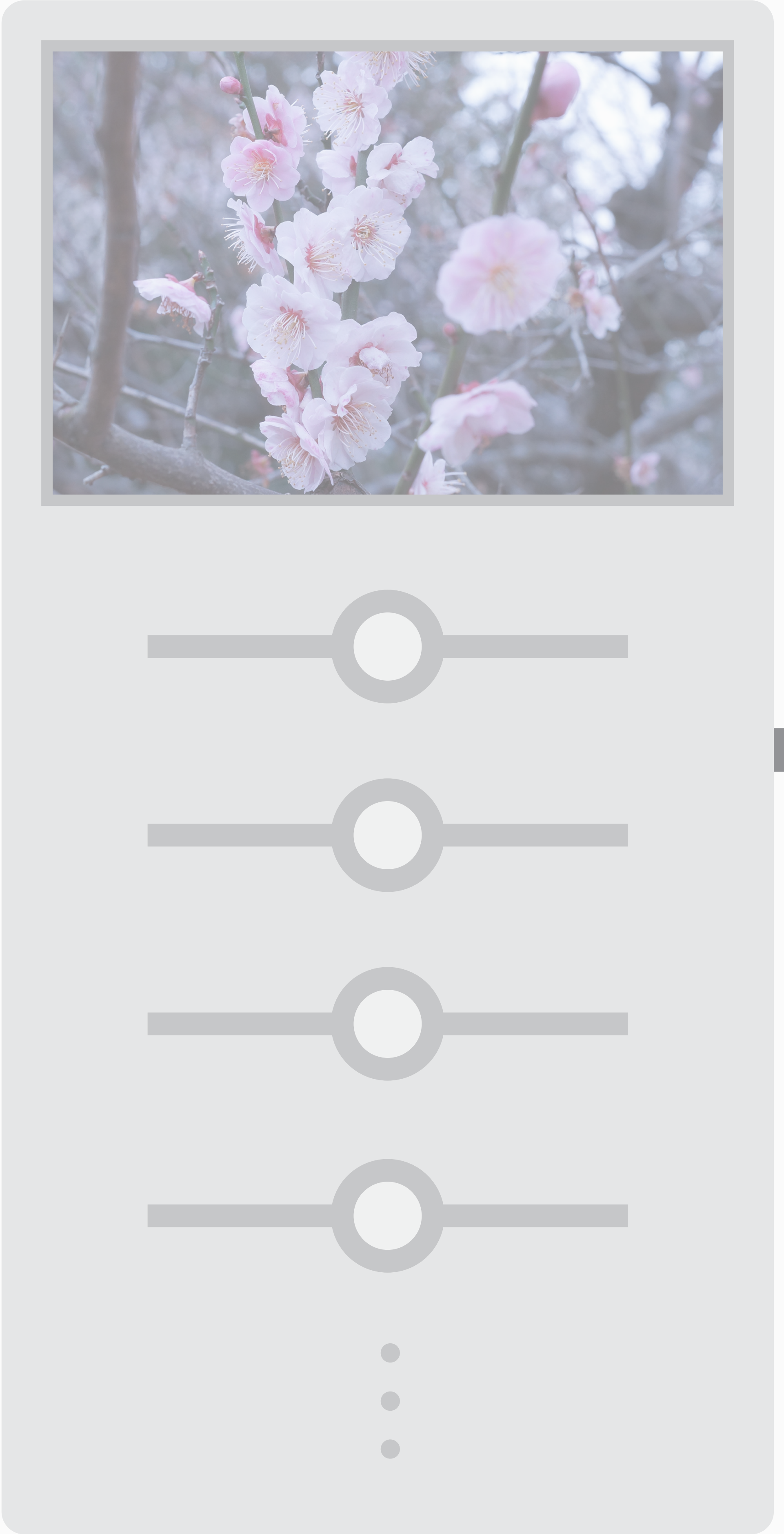
Output:
An optimal
parameter set

Target:
n design parameters
(e.g., photo enhance)

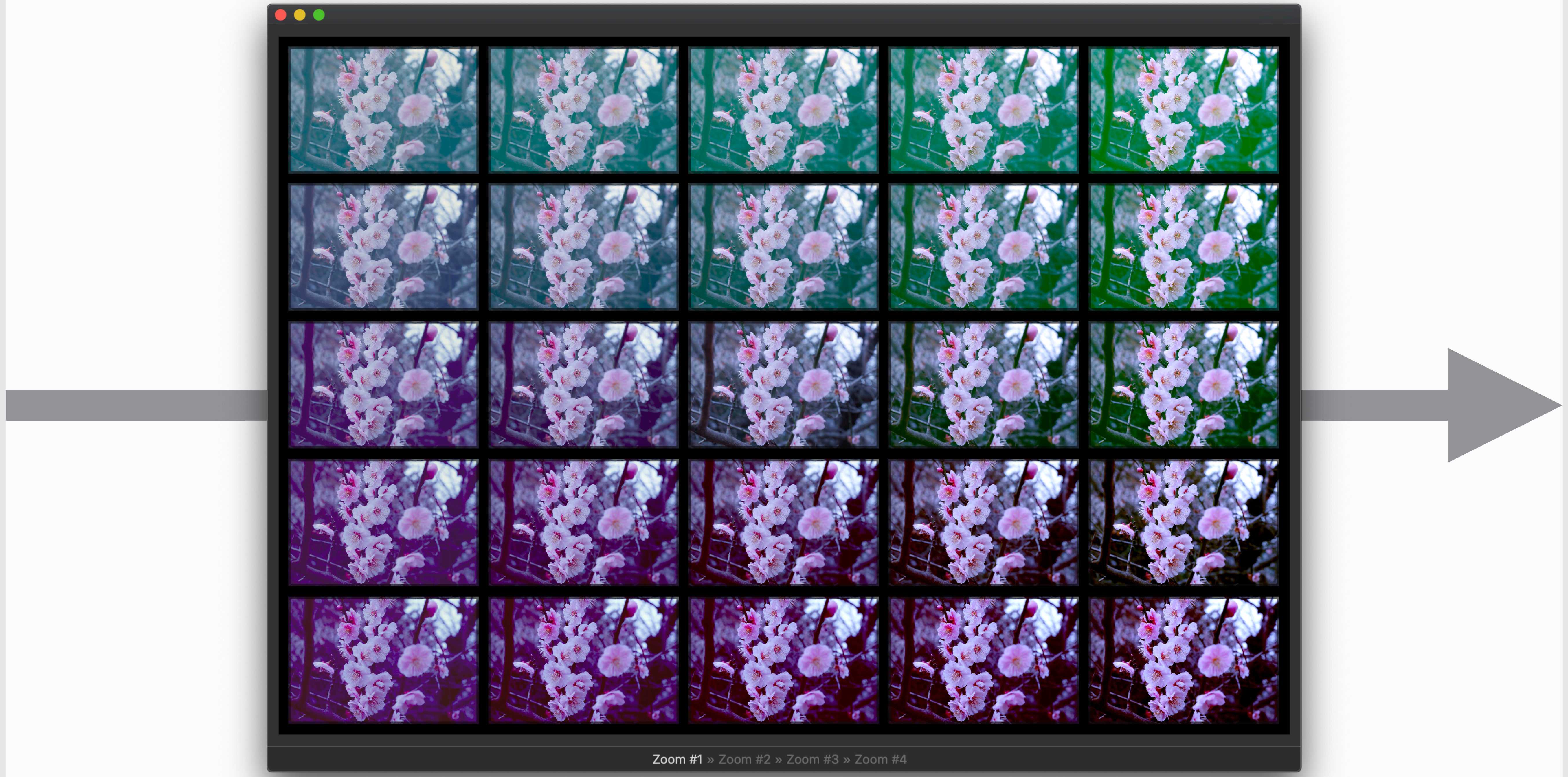
Sequential Gallery:
An interactive optimization framework
where the user sequentially performs
2D search subtasks via a grid interface

Output:
An optimal
parameter set

Target:
n design parameters
(e.g., photo enhance)

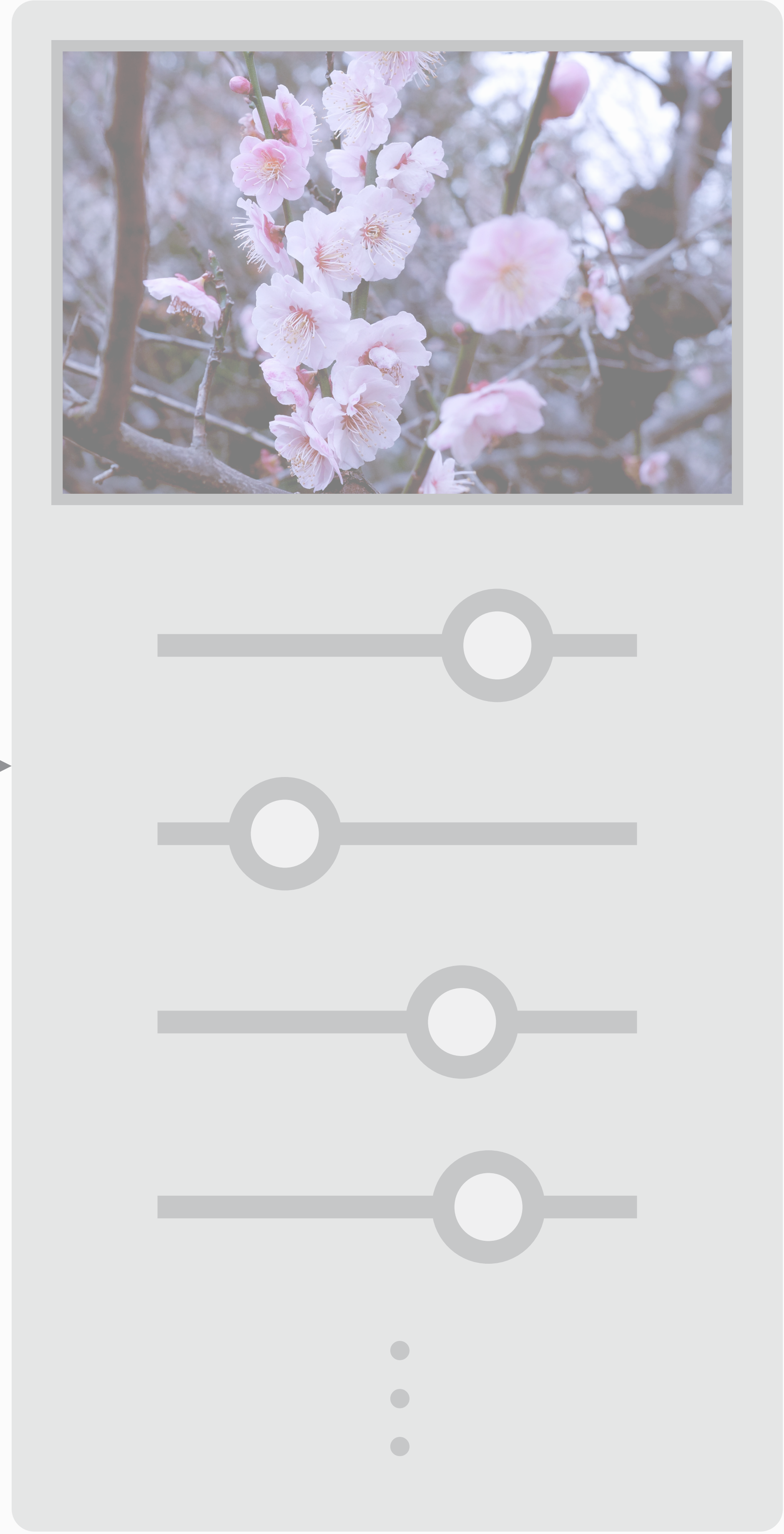


Sequential Gallery:
An interactive optimization framework where the user sequentially performs 2D search subtasks via a grid interface



2D search subtask #1

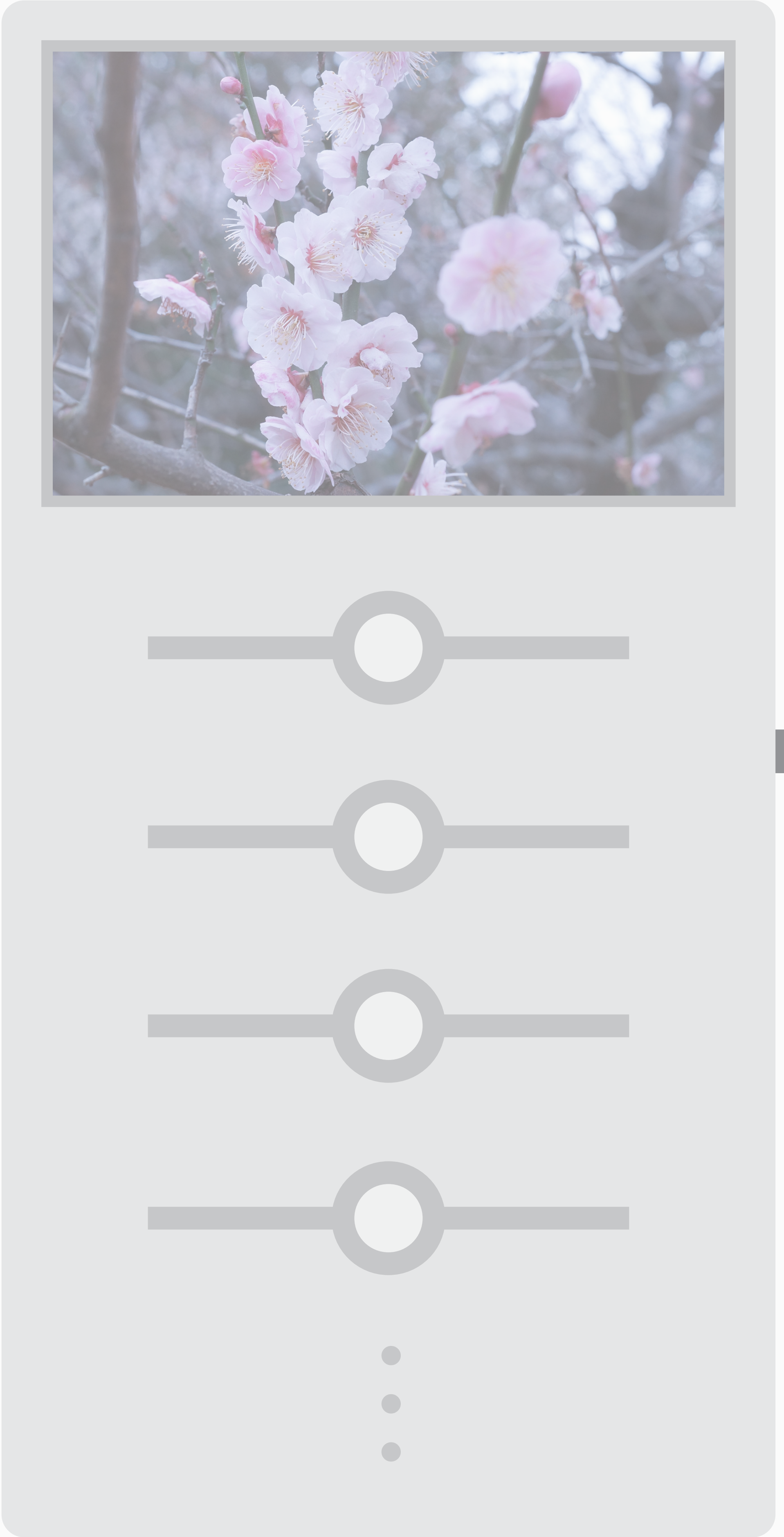
Output:
An optimal parameter set

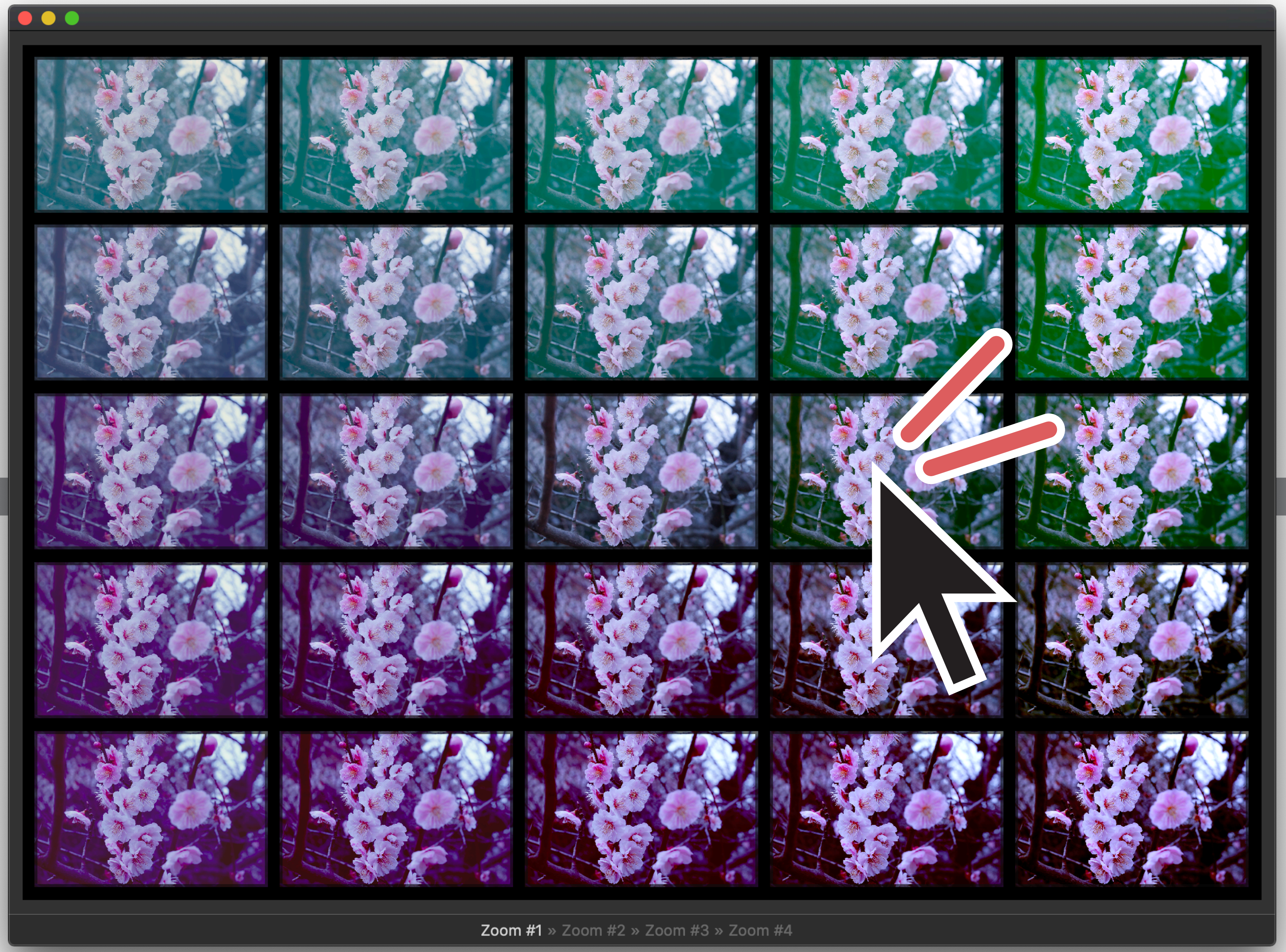


Target:
n design parameters
(e.g., photo enhance)

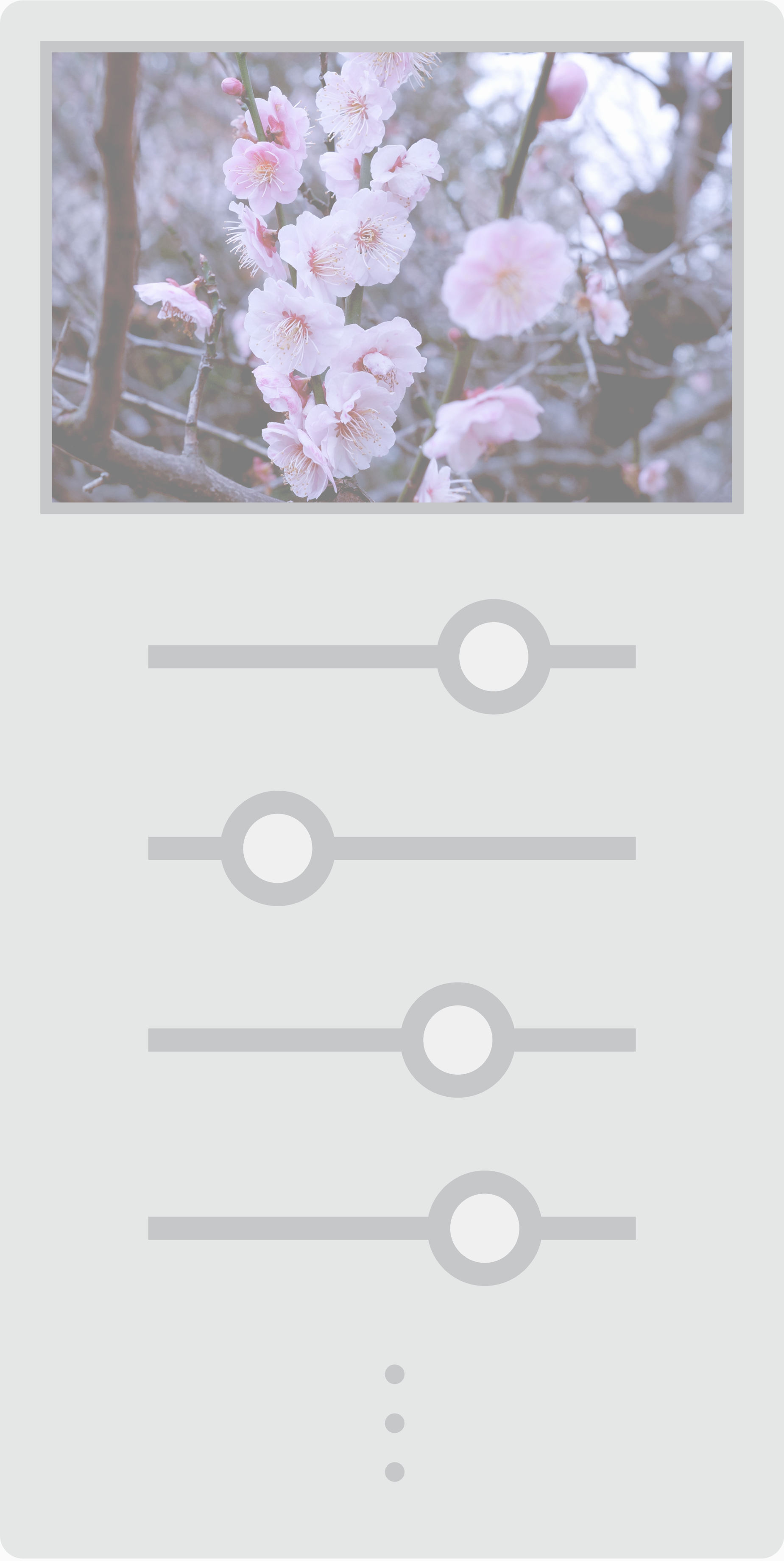
Sequential Gallery:
An interactive optimization framework
where the user sequentially performs
2D search subtasks via a grid interface

Output:
An optimal
parameter set





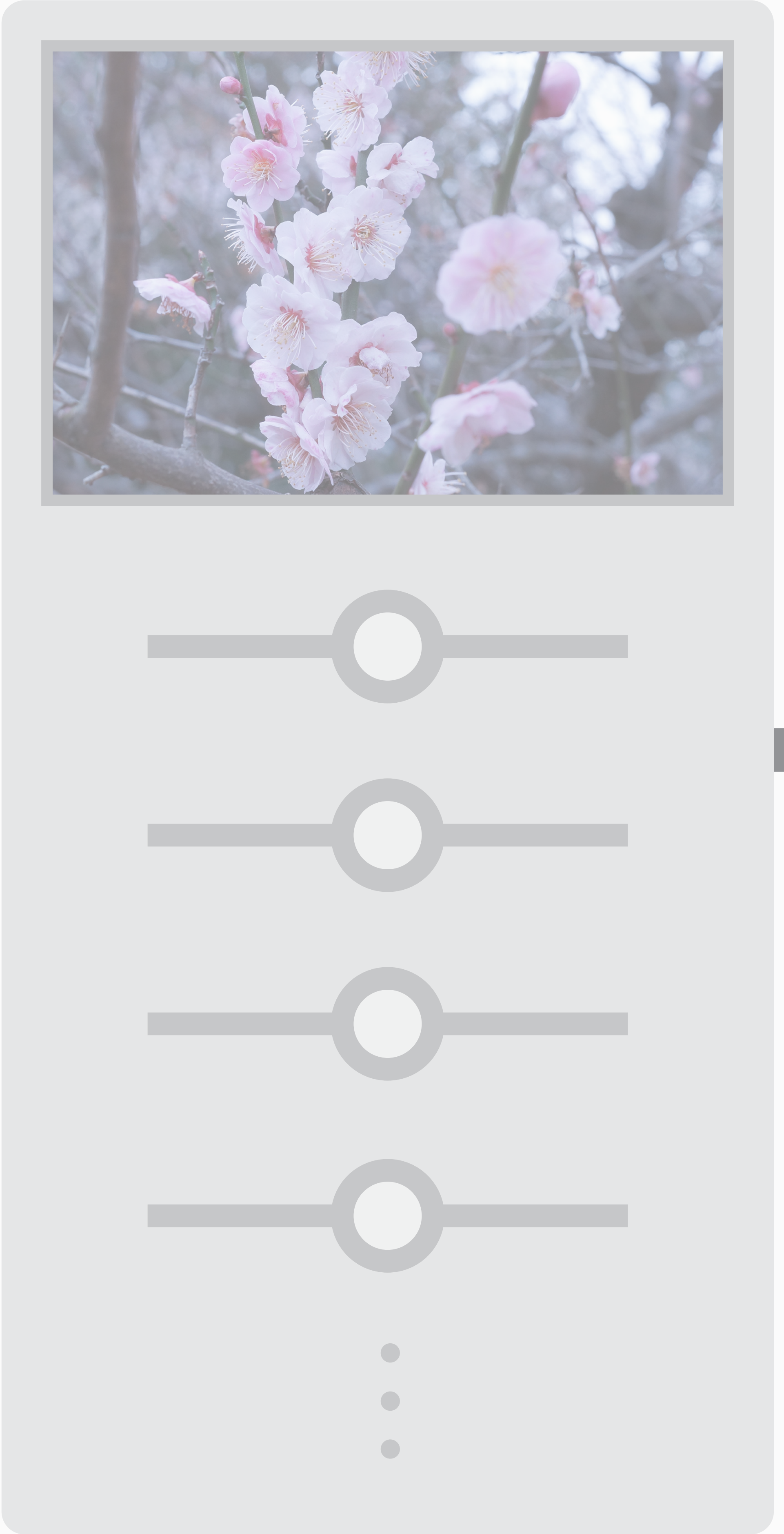
2D search subtask #1

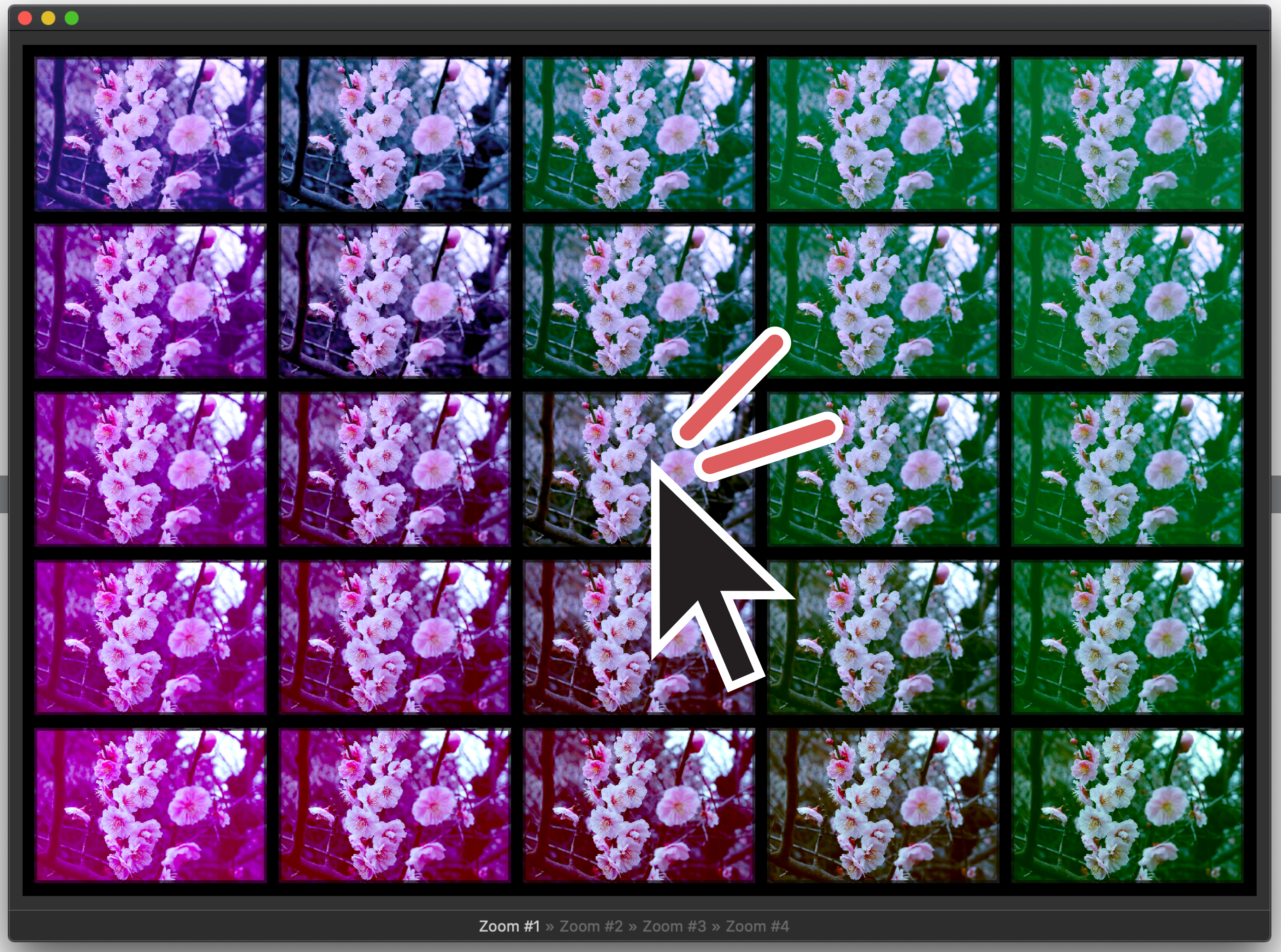


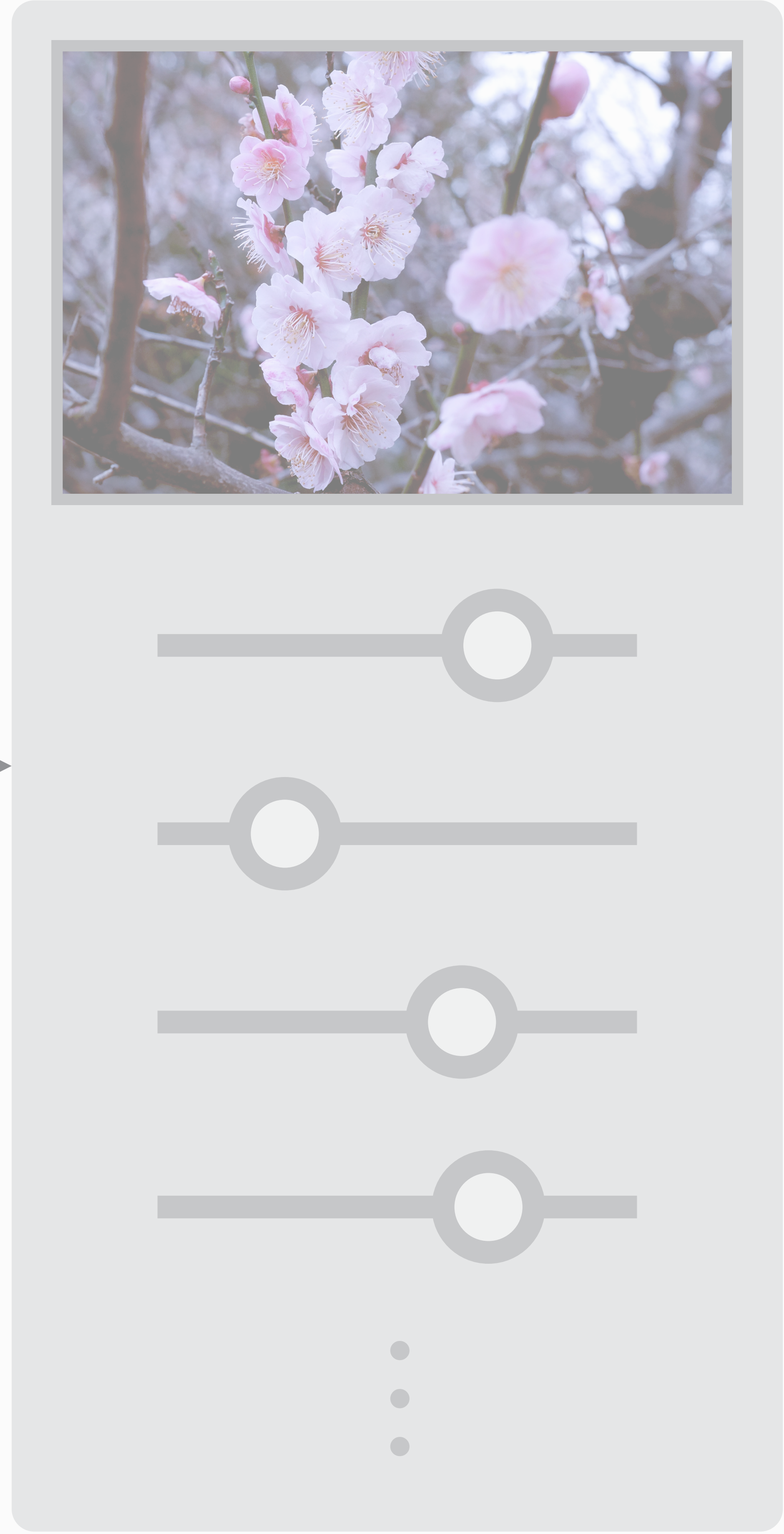
Target:
n design parameters
(e.g., photo enhance)

Sequential Gallery:
An interactive optimization framework
where the user sequentially performs
2D search subtasks via a grid interface

Output:
An optimal
parameter set





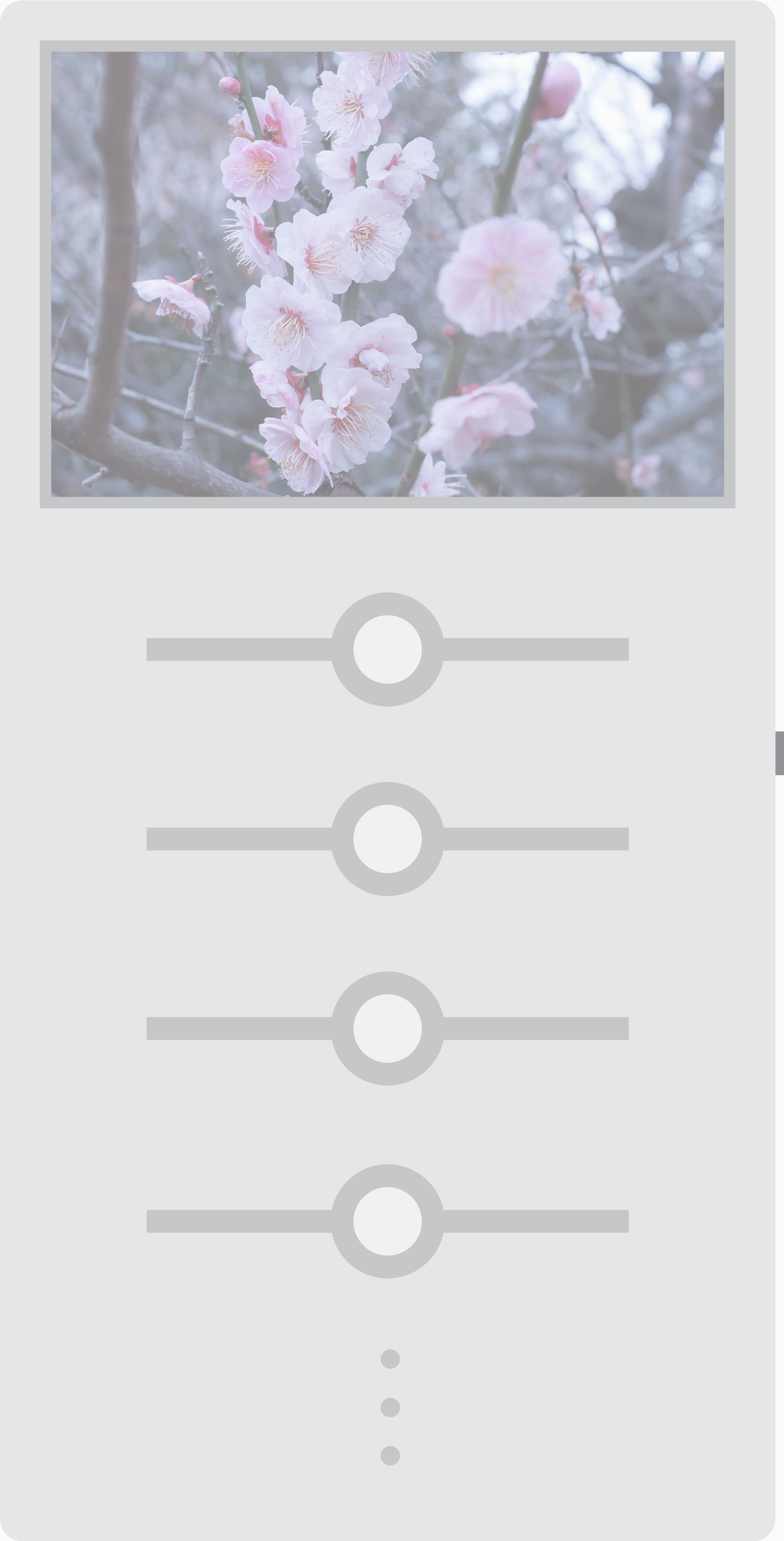


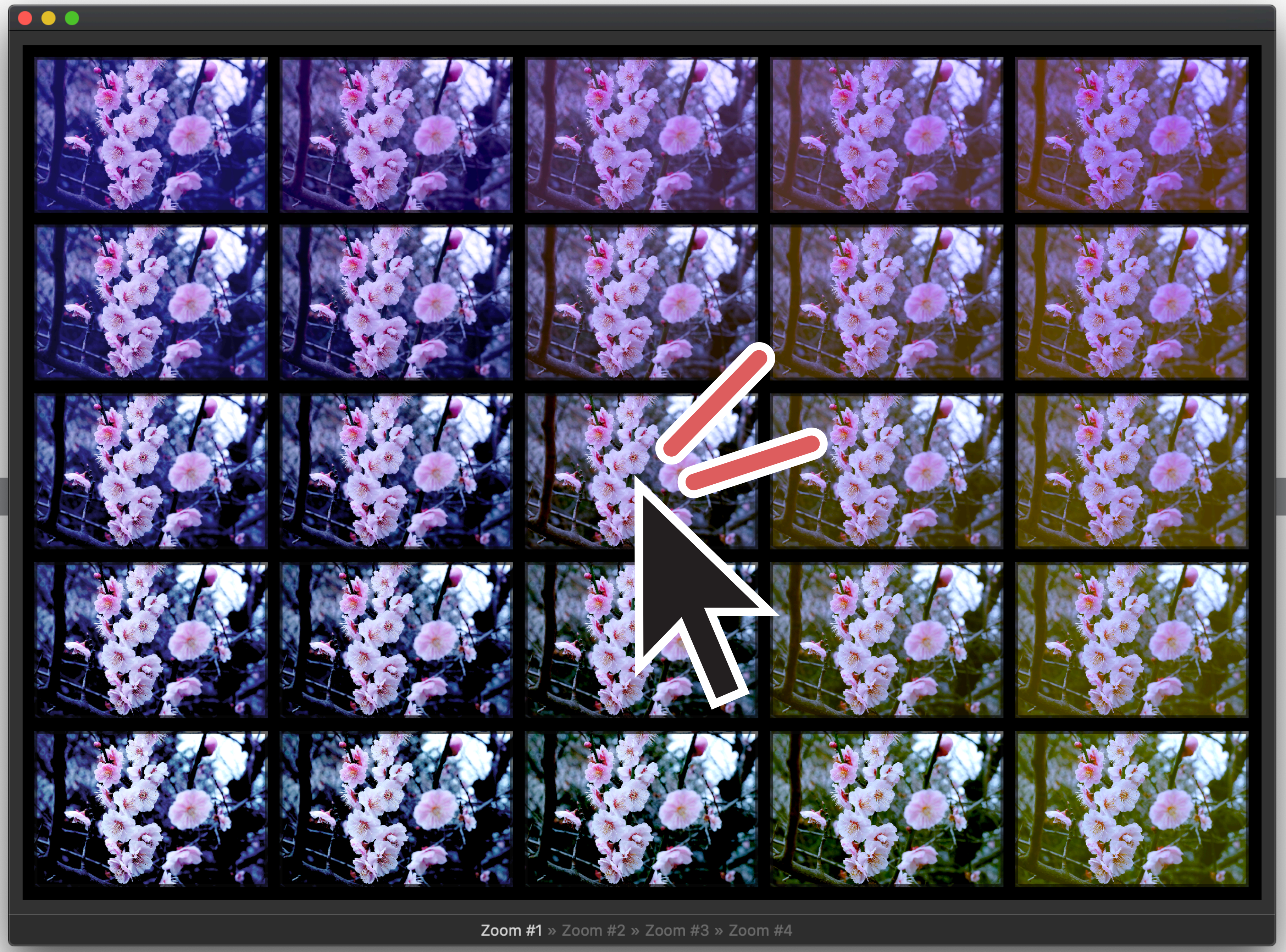
2D search subtask #2

Target:
n design parameters
(e.g., photo enhance)

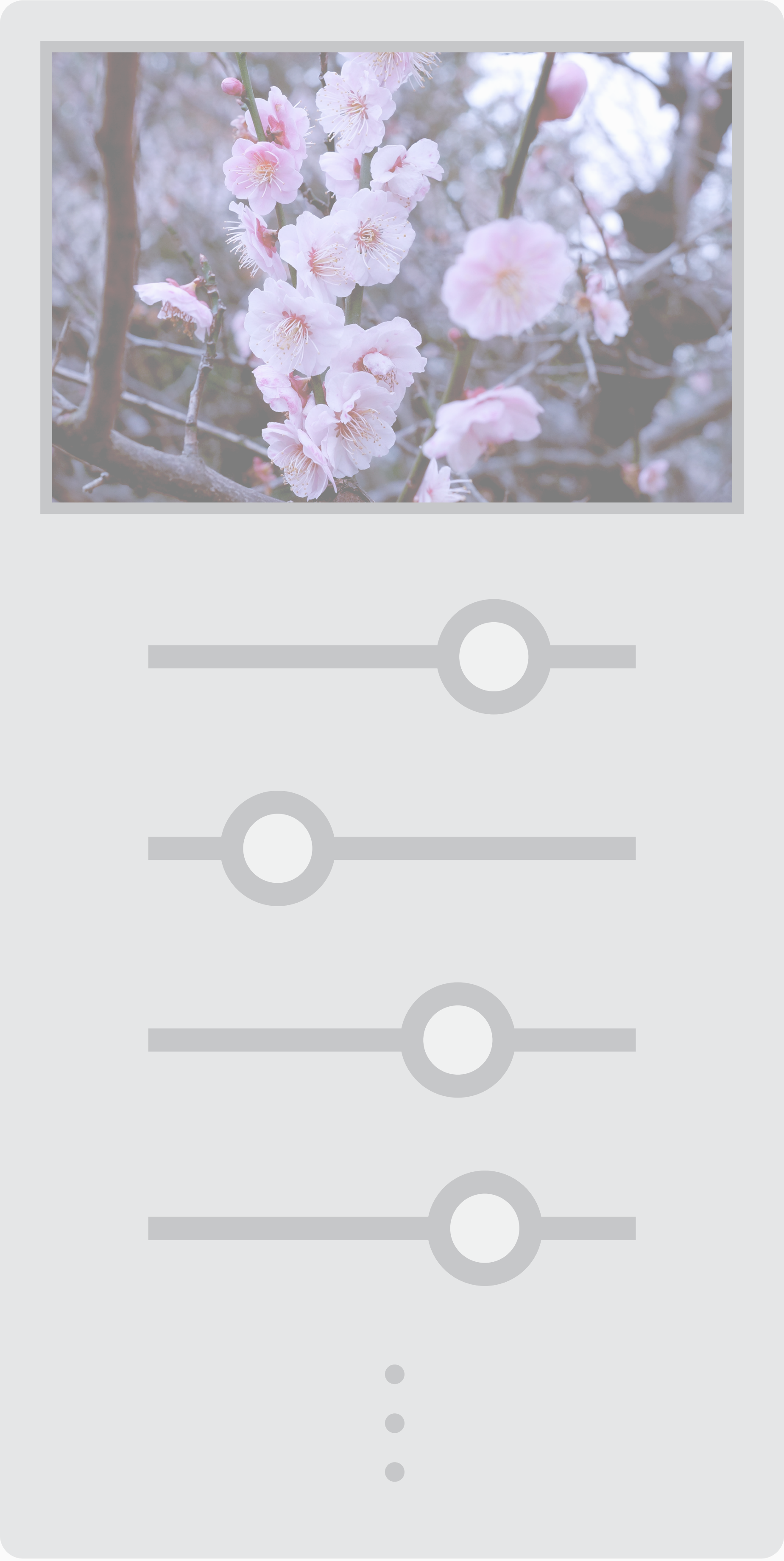
Sequential Gallery:
An interactive optimization framework where the user sequentially performs 2D search subtasks via a grid interface

Output:
An optimal parameter set





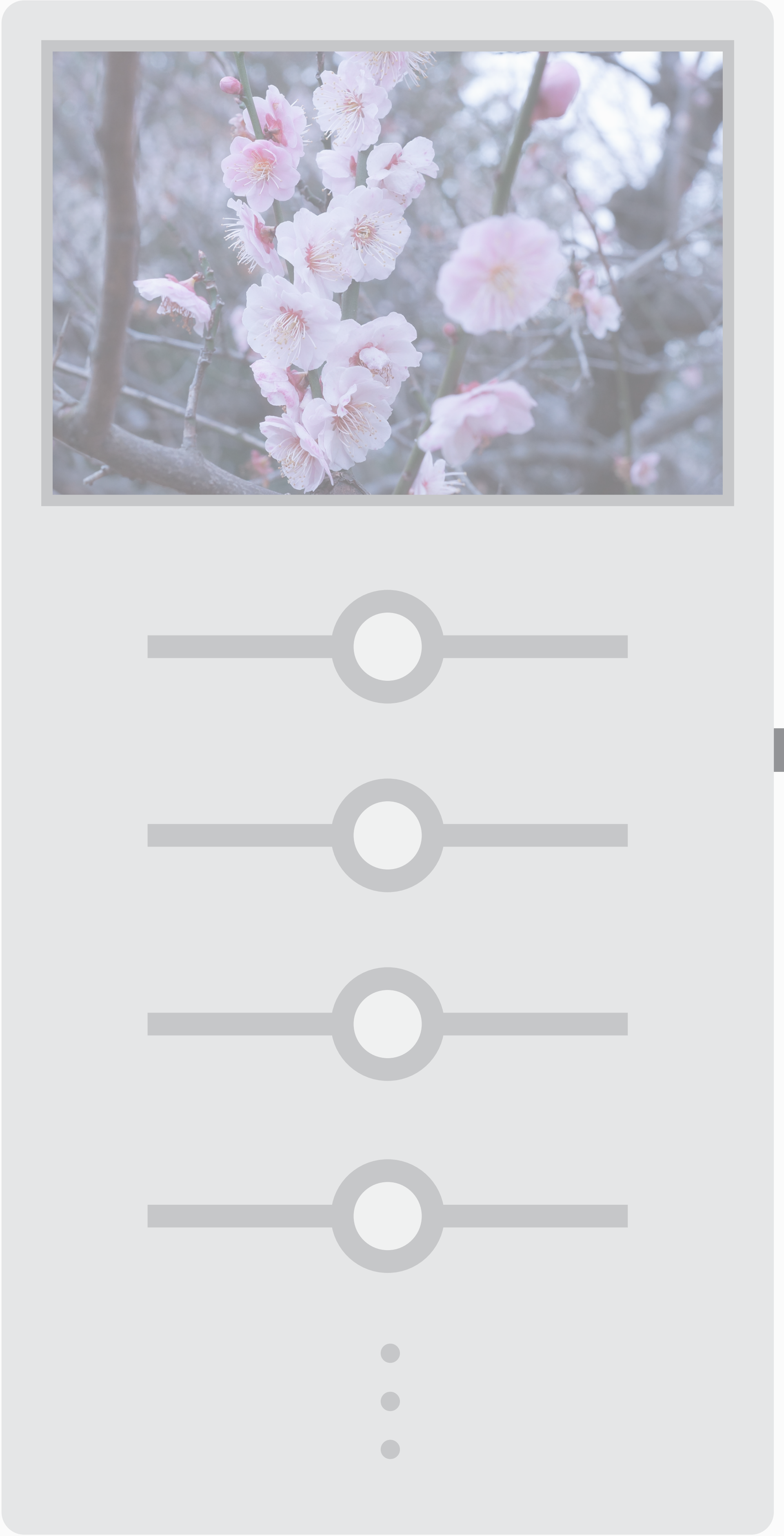
2D search subtask #3



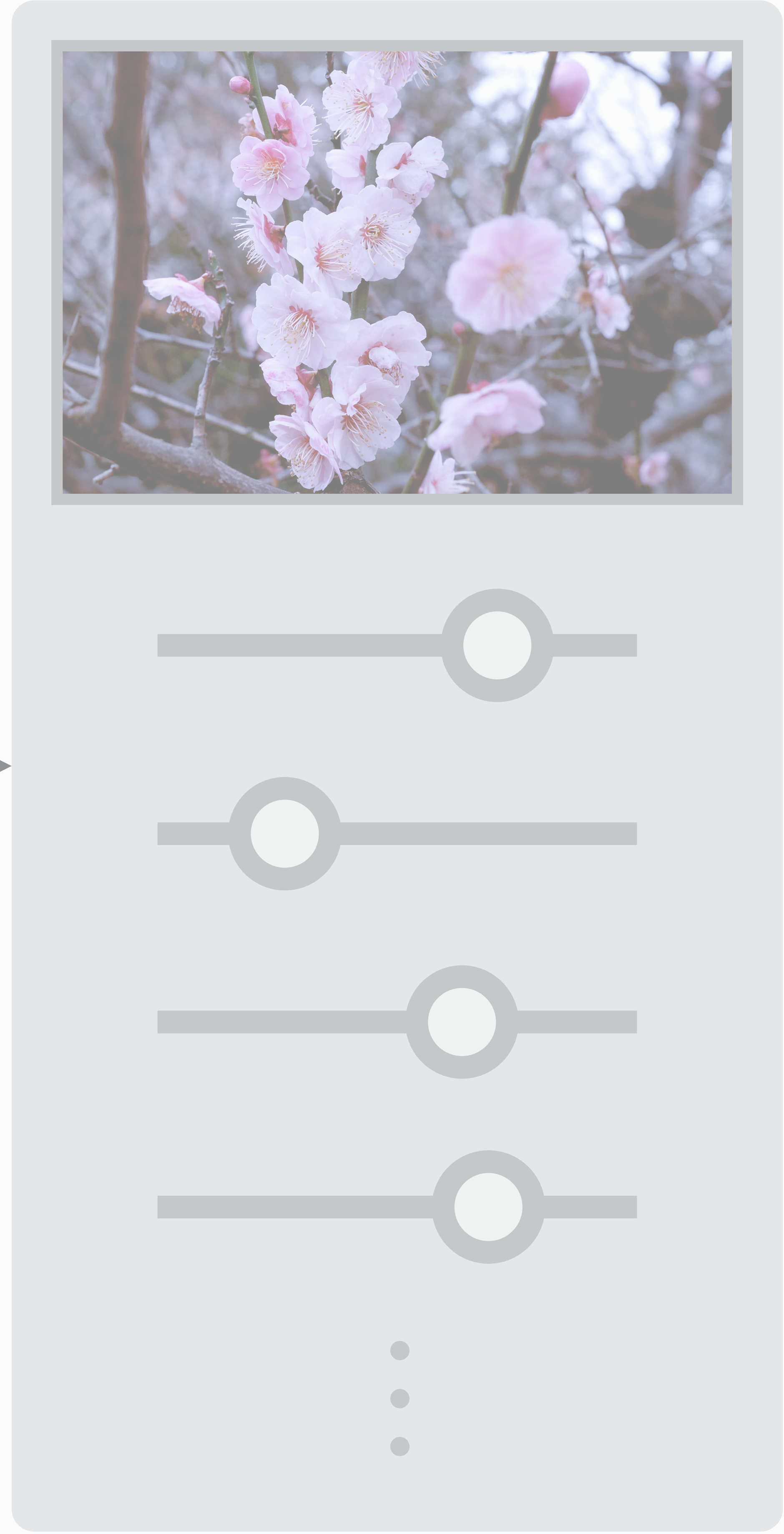
Target:
n design parameters
(e.g., photo enhance)

Sequential Gallery:
An interactive optimization framework
where the user sequentially performs
2D search subtasks via a grid interface

Output:
An optimal
parameter set



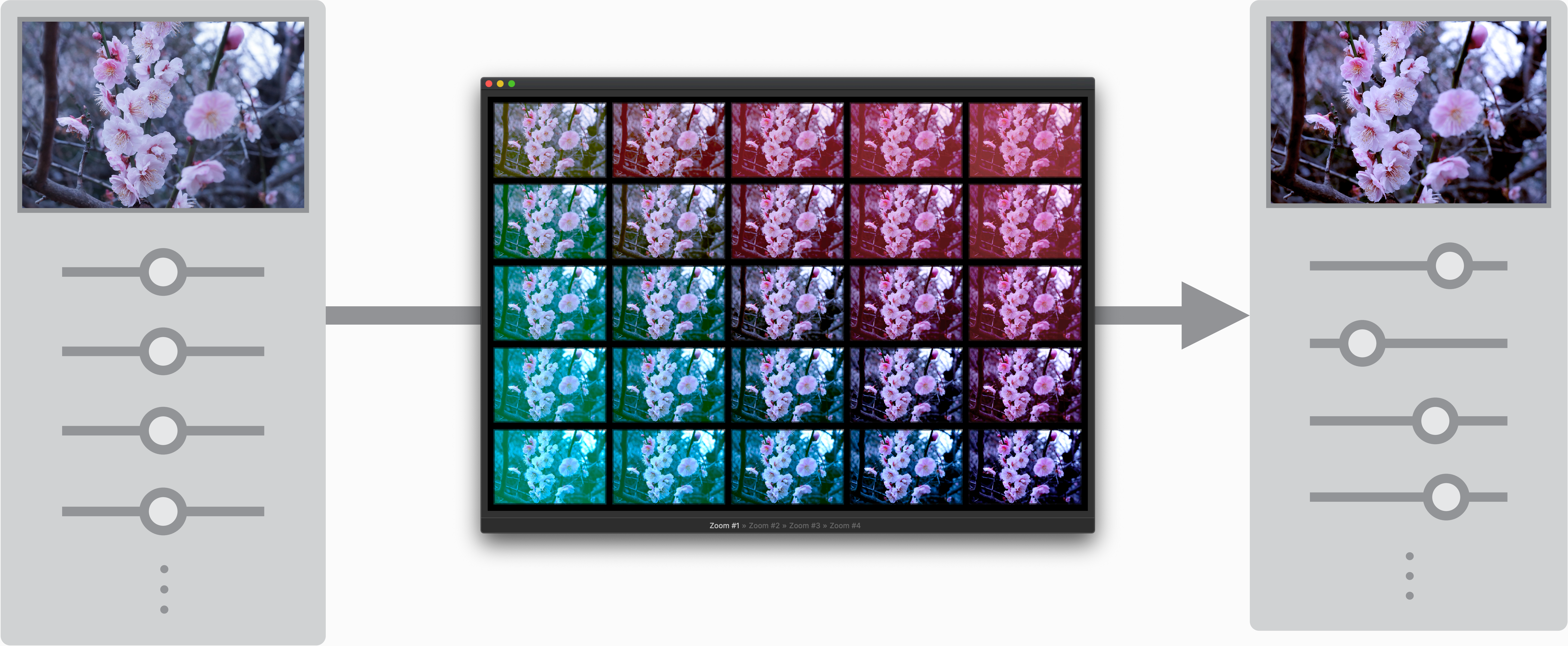
2D search subtask #4



Target:
n design parameters
(e.g., photo enhance)

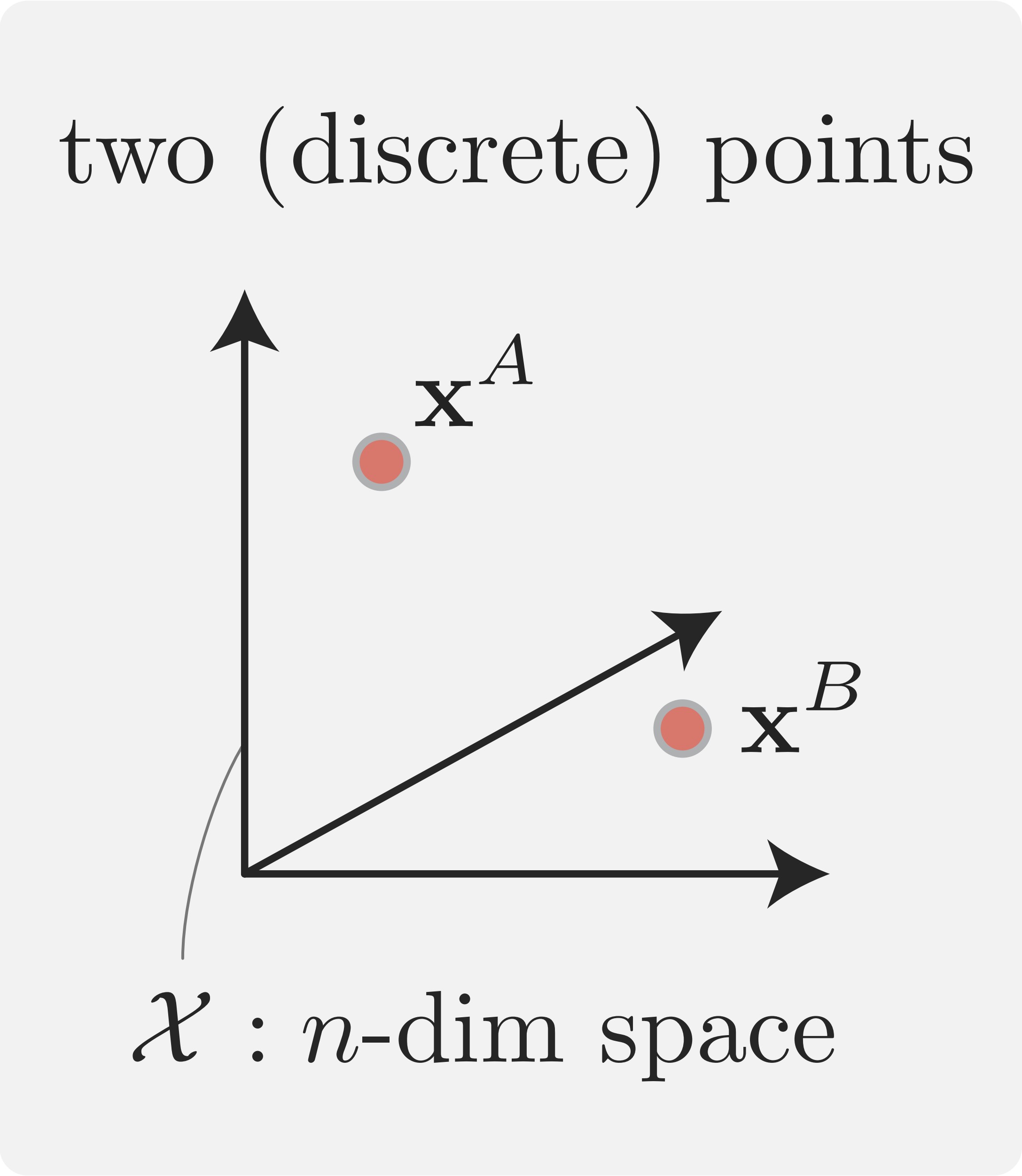
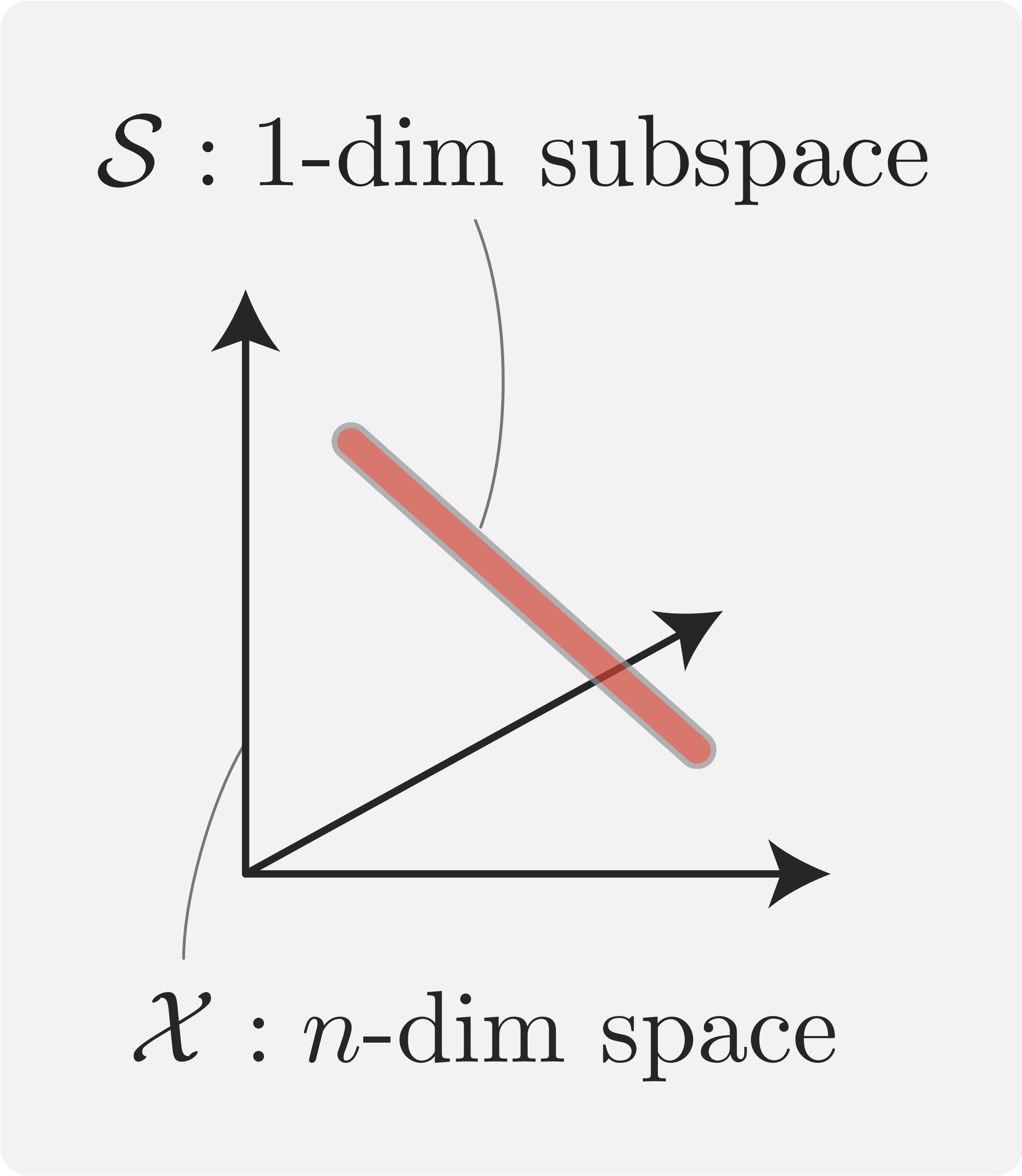
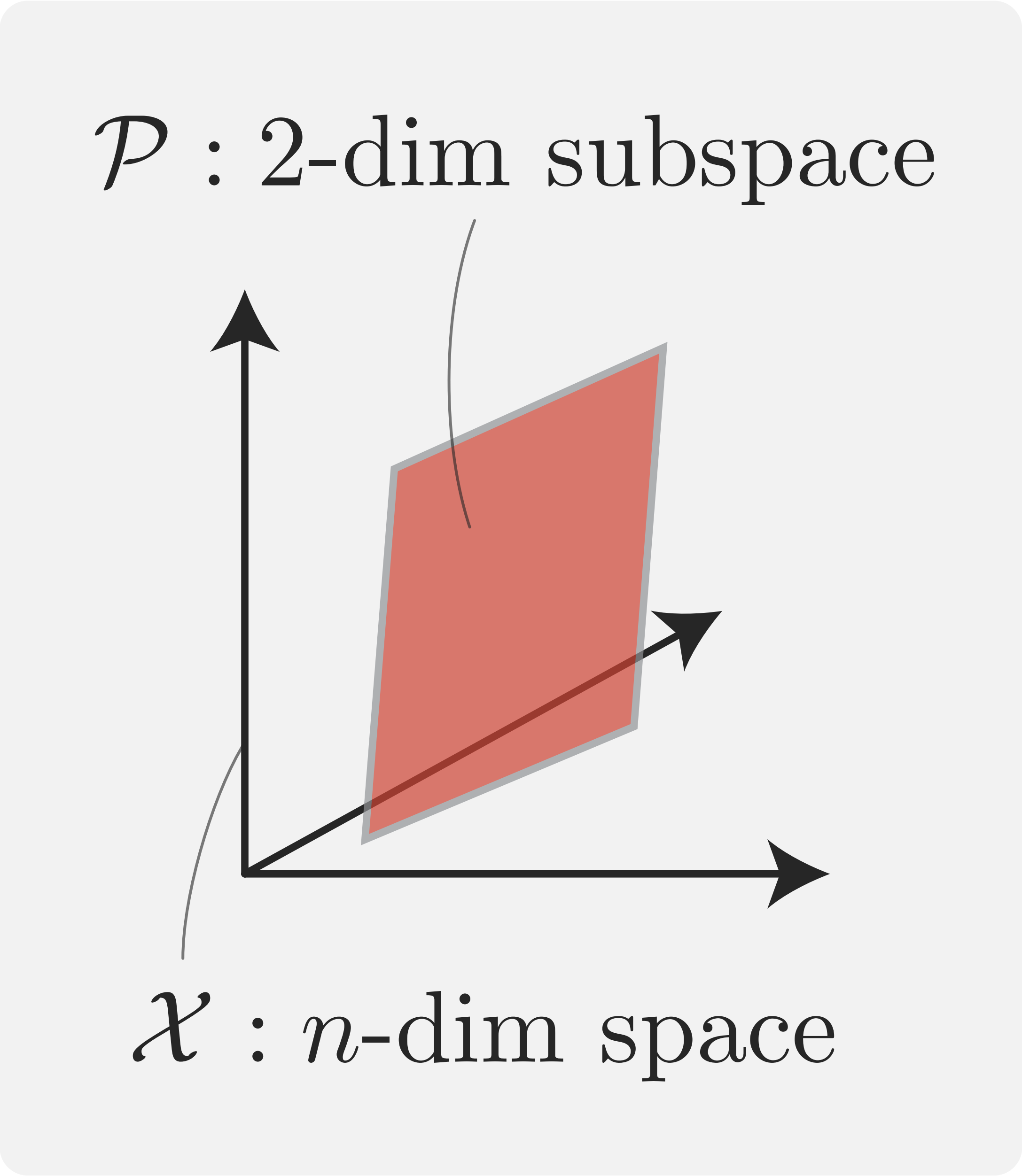
Sequential Gallery:
An interactive optimization framework
where the user sequentially performs
2D search subtasks via a grid interface

Output:
An optimal
parameter set

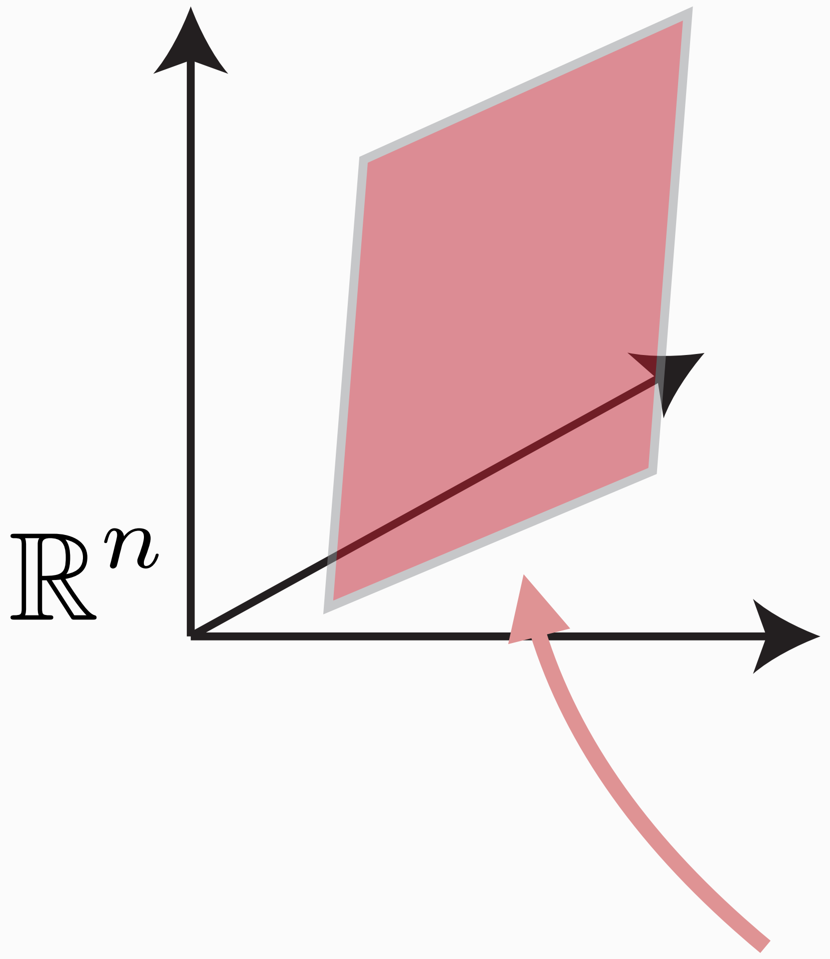


Query Design

Plane Search Query

	[Brochu+, NIPS 2007] **“Pairwise Comparison” Query**	**[Koyama+, SIGGRAPH 2017]** **“Line Search” Query** **(Sequential Line Search)**	**[Ours]** **“Plane Search” Query** **(Sequential Plane Search)**
Query type:			
User task:	$\mathbf{x}^{\text{chosen}} = \underset{\mathbf{x} \in \{\mathbf{x}^A, \mathbf{x}^B\}}{\operatorname{argmax}} \, g(\mathbf{x})$	$\mathbf{x}^{\text{chosen}} = \underset{\mathbf{x} \in \mathcal{S}}{\operatorname{argmax}} \, g(\mathbf{x})$	$\mathbf{x}^{\text{chosen}} = \underset{\mathbf{x} \in \mathscr{P}}{\operatorname{argmax}} \, g(\mathbf{x})$
Notes:	Is the first PBO algorithm	Needs fewer iterations than [Brochu+ NIPS 2007]	Needs even fewer iterations, and has a good compatibility with grid interfaces

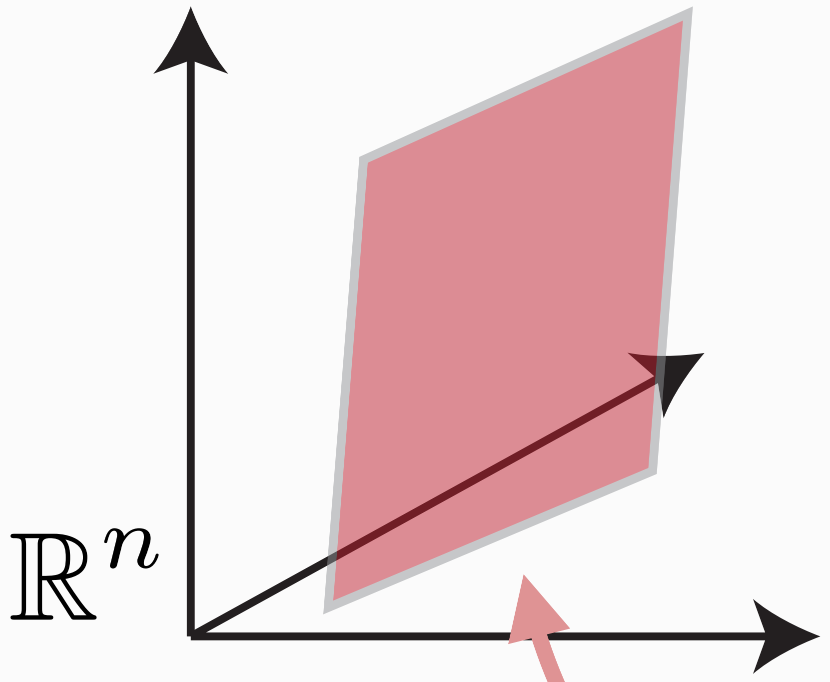
Sequential Gallery Workflow



2-dimensional search subspaces (= **search planes**)
determined by the **sequential plane search** algorithm

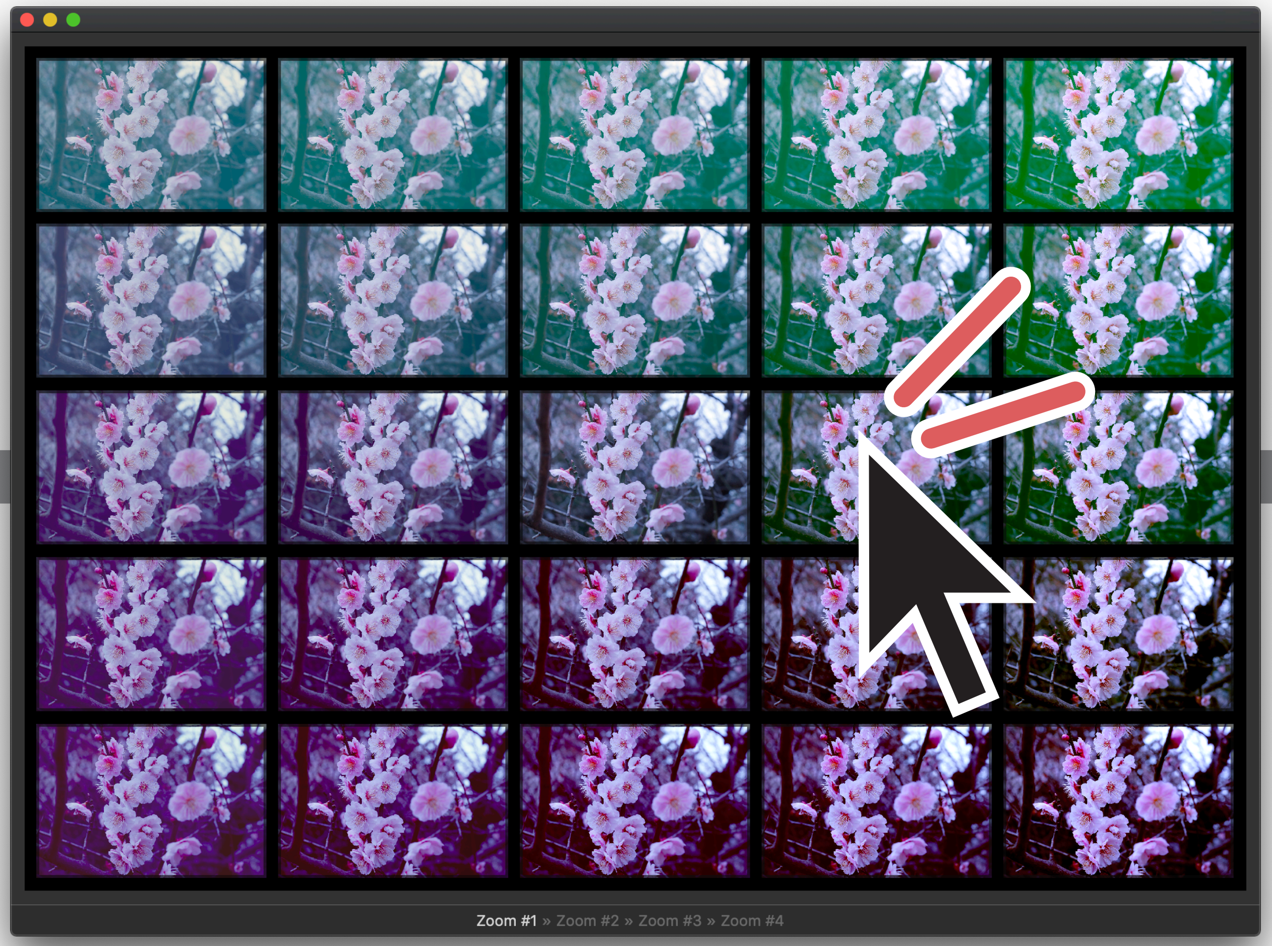
2-dimensional search subspaces (= **search planes**)
determined by the **sequential plane search** algorithm

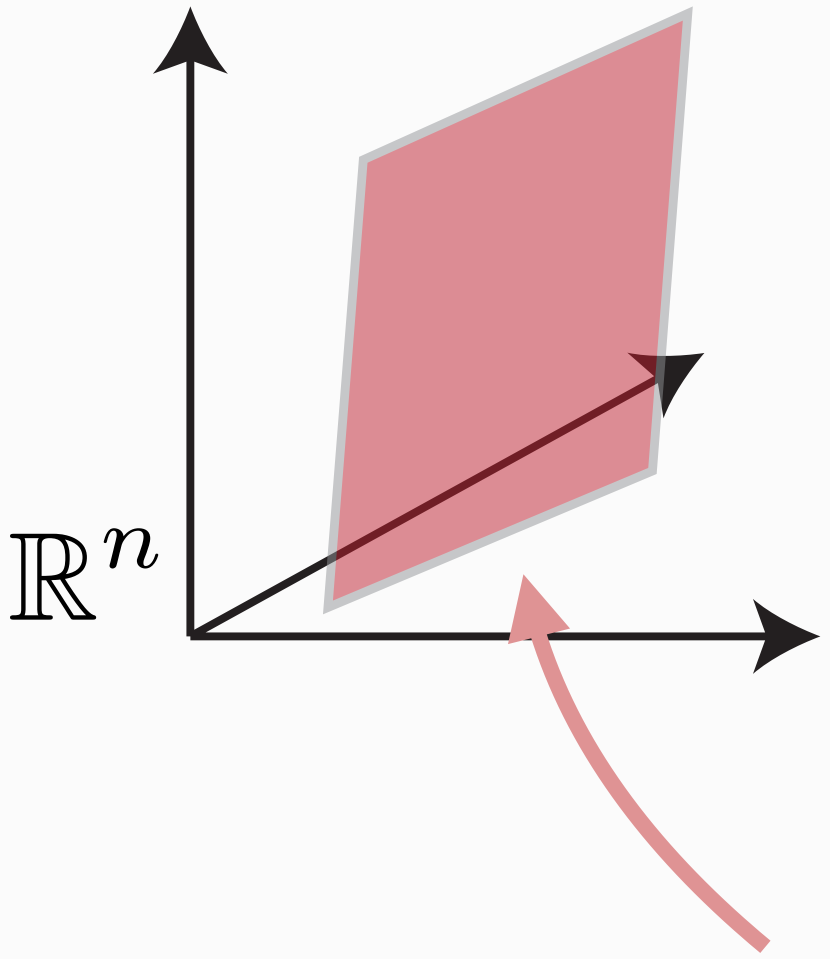
…　2D search subtask



2-dimensional search subspaces (= **search planes**)
determined by the **sequential plane search** algorithm

… 2D search subtask





2-dimensional search subspaces (= **search planes**)
determined by the **sequential plane search** algorithm

Yuki Koyama, Issei Sato, and Masataka Goto. Sequential Gallery for Interactive Visual Design Optimization. ACM Trans. Graph. (SIGGRAPH 2020)

2-dimensional search subspaces (= **search planes**)
determined by the **sequential plane search** algorithm

… 2D search subtask

2-dimensional search subspaces (= **search planes**)
determined by the **sequential plane search** algorithm

... 2D search subtask

2-dimensional search subspaces (= **search planes**)
determined by the **sequential plane search** algorithm

2-dimensional search subspaces (= **search planes**)
determined by the **sequential plane search** algorithm

2-dimensional search subspaces (= **search planes**)
determined by the **sequential plane search** algorithm

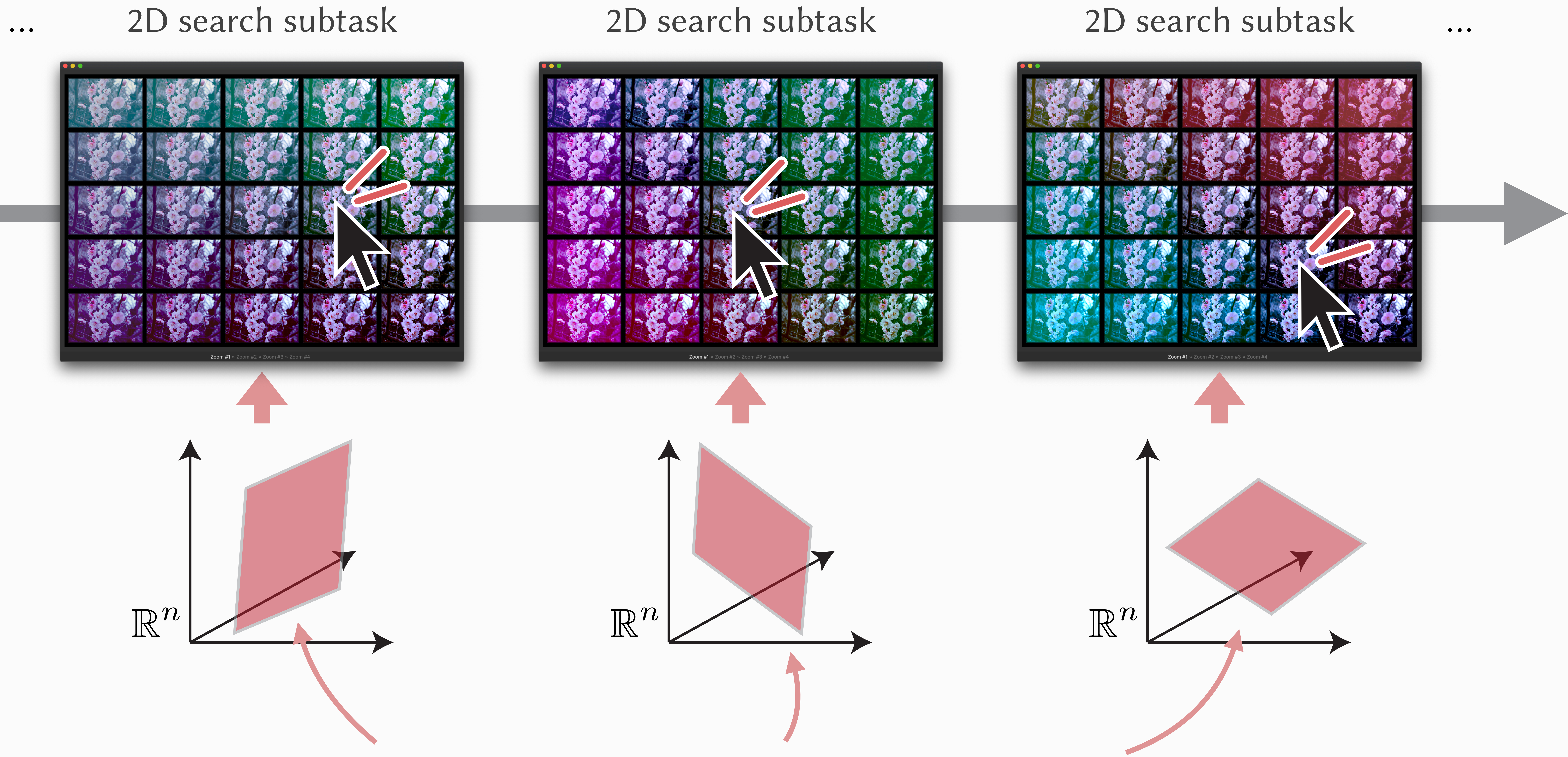

...
2D search subtask
2D search subtask
User's feedback
Next search plane
$\mathbb{R}^n$
$\mathbb{R}^n$
$\mathbb{R}^n$
2-dimensional search subspaces (= **search planes**)
determined by the **sequential plane search** algorithm

... 2D search subtask 2D search subtask

2-dimensional search subspaces (= **search planes**)
determined by the **sequential plane search** algorithm

2-dimensional search subspaces (= **search planes**)
determined by the **sequential plane search** algorithm

2-dimensional search subspaces (= **search planes**)
determined by the **sequential plane search** algorithm



2-dimensional search subspaces (= **search planes**)
determined by the **sequential plane search** algorithm

Benefits of Grid Gallery Interface

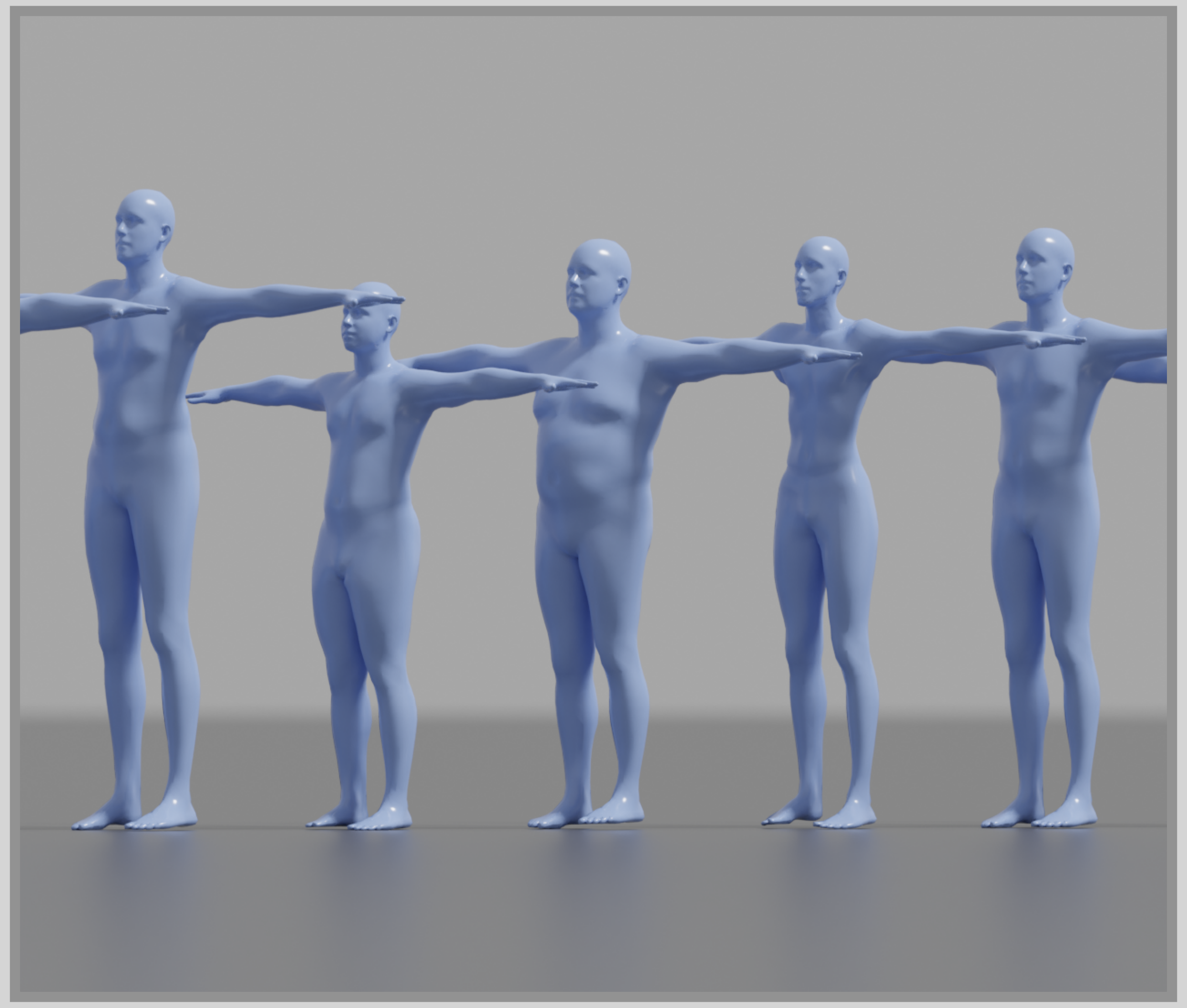
- **Allows the user to easily grasp the available options** in the 2D subspace by just seeing the grid view
- **WYSIWYG** (What-You-See-Is-What-You-Get); do not need to be aware of raw parameters
- Compatibility with the sequential-plane-search task (i.e., 2D search)

Applications:
Possible Scenarios and Demonstrations

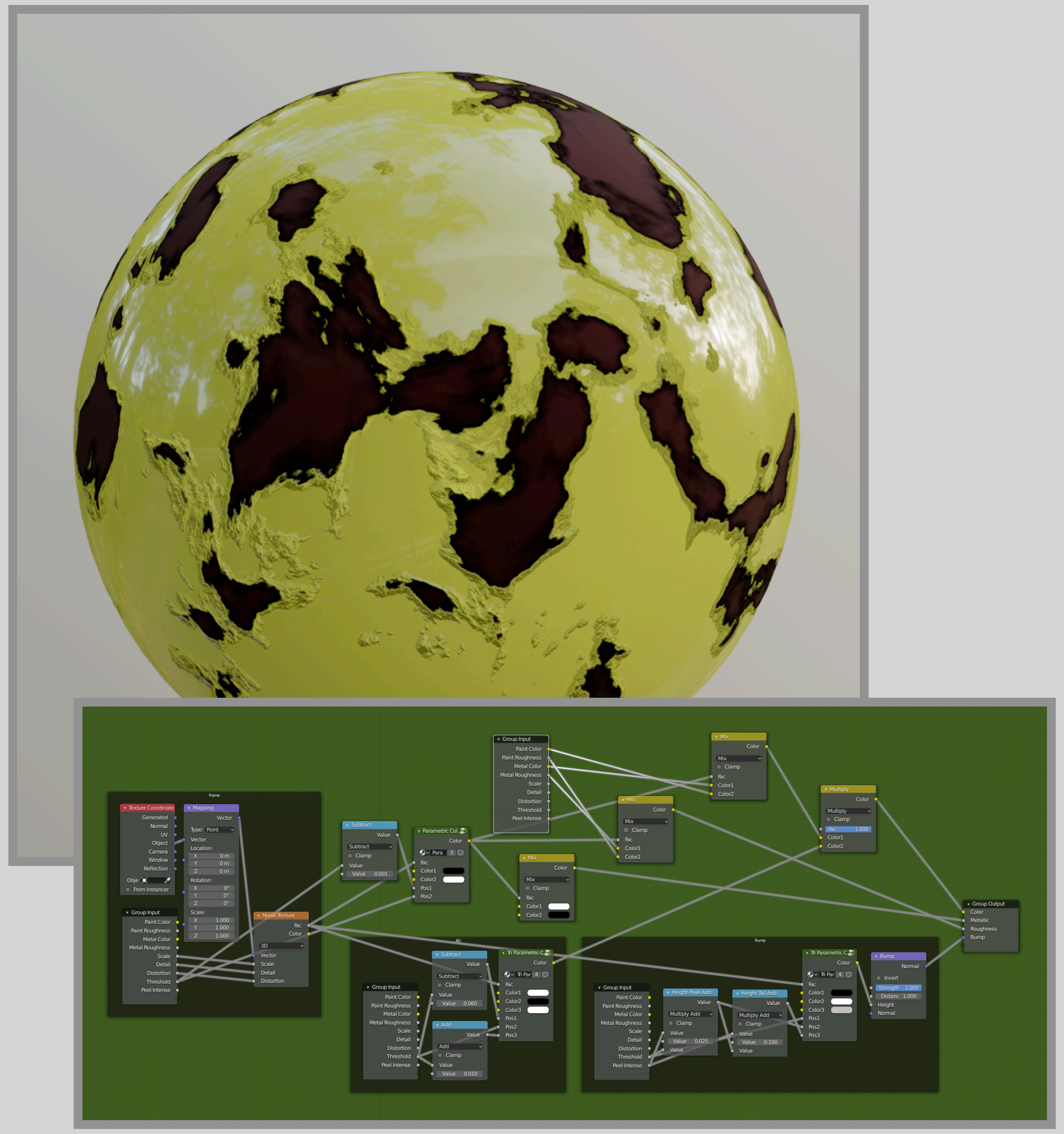
Potential Applications

Photo color enhancement

Generative modeling



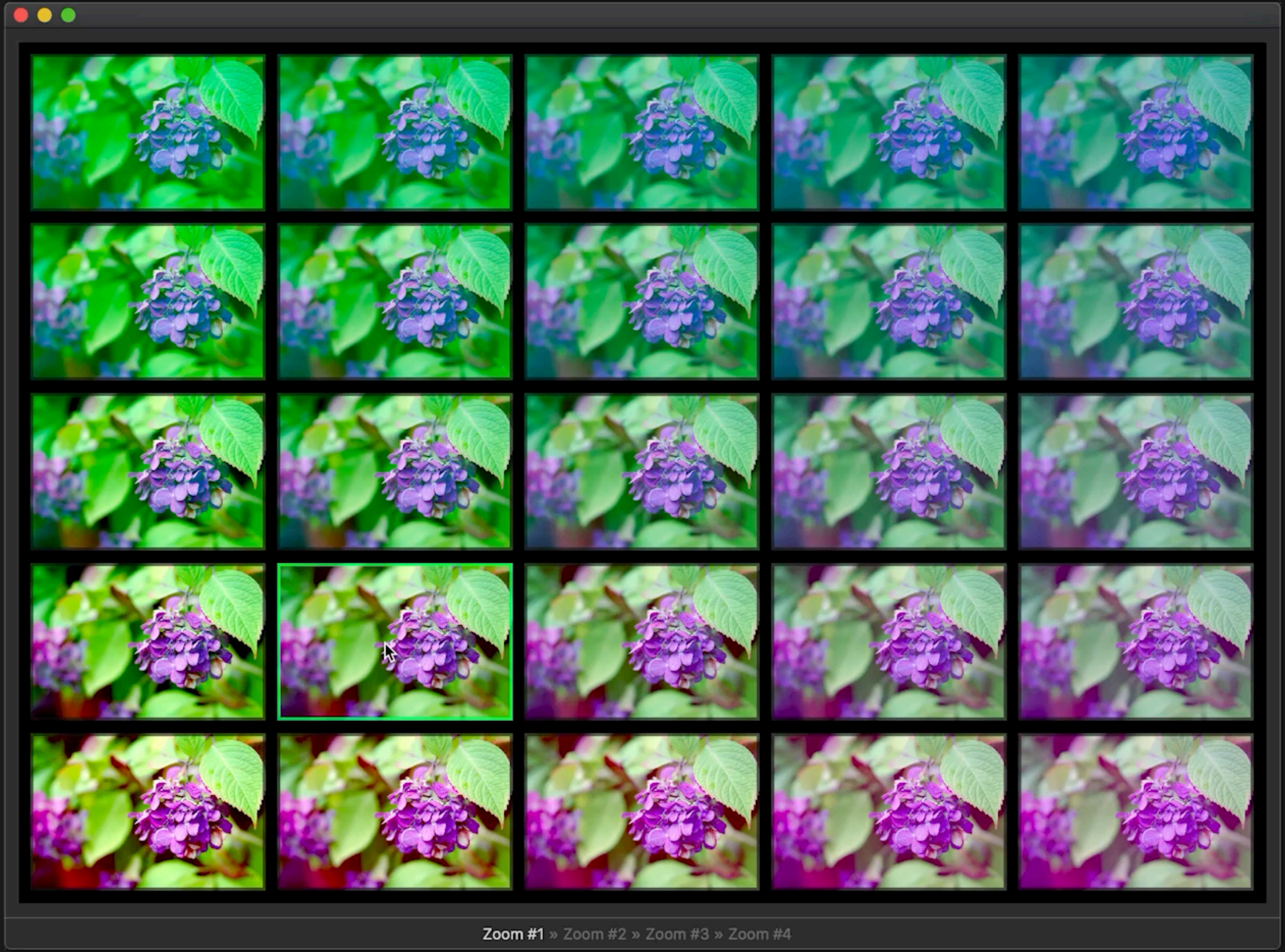
Procedural texturing



… and many other parametric design scenarios

Photo Color Enhancement (12D)

Brightness, contrast, saturation, shadows (RGB), midtones (RGB), and highlights (RGB)



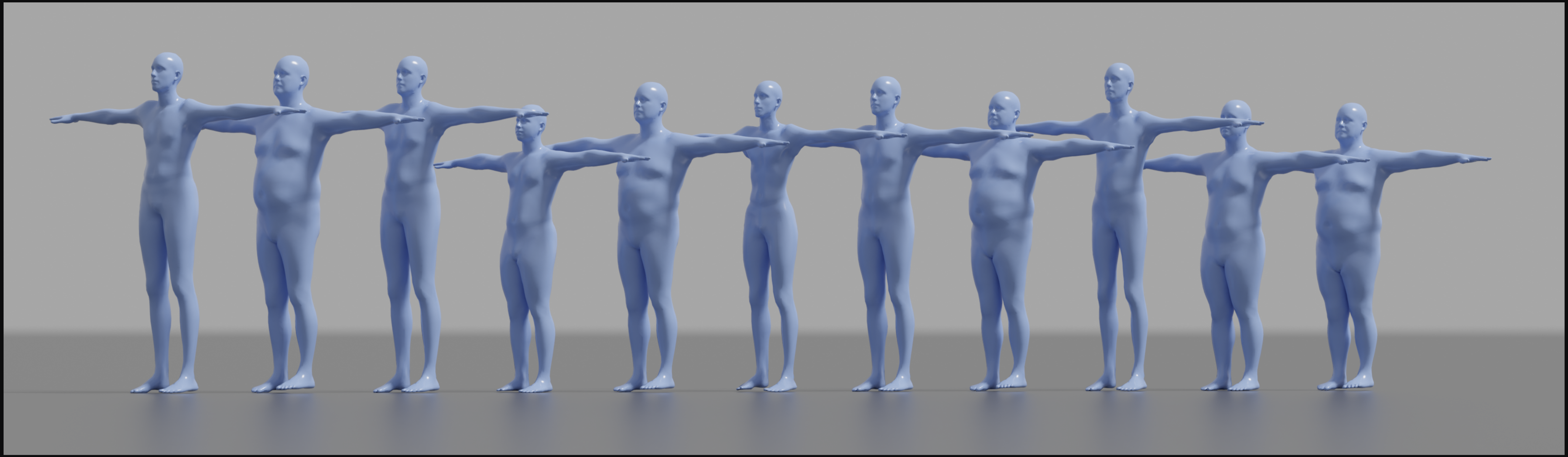
x1.5 speed

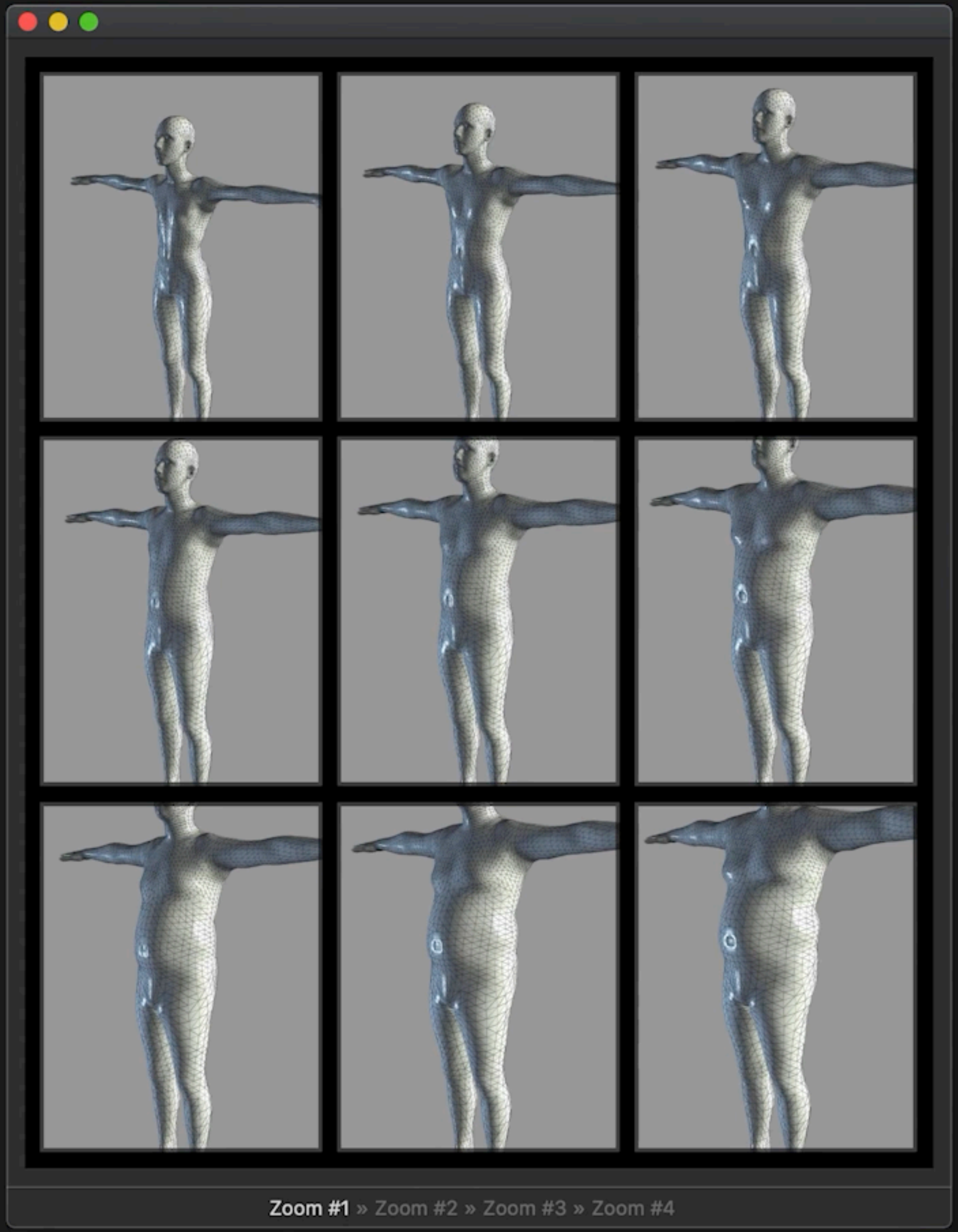
Original photograph

Enhanced photograph
(after 4 iterations)

Body Shaping (10D)

Using the SMPL model [Loper+15] (the first 10 principal components)





Goal: Body shaping from a character description

"He was of medium height, solidly built, wide in the shoulders, thick in the neck, with a jovial heavy-jawed red face [...]"

Dashiell Hammett. 1930. *The Maltese Falcon.*

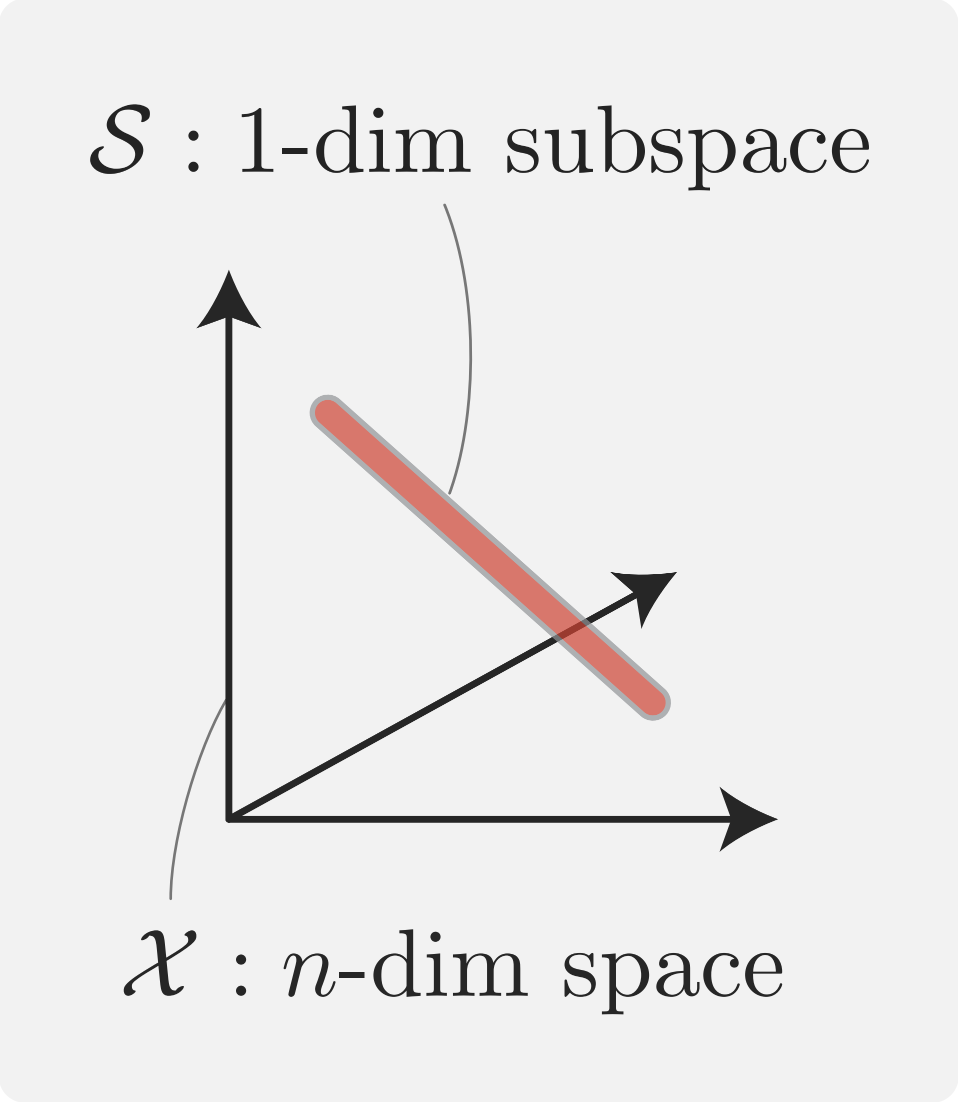
"He was of medium height, solidly built, wide in the shoulders, thick in the neck, with a jovial heavy-jawed red face [...]"

Dashiell Hammett. 1930. *The Maltese Falcon.*

Evaluation:
Optimization Performance Comparison

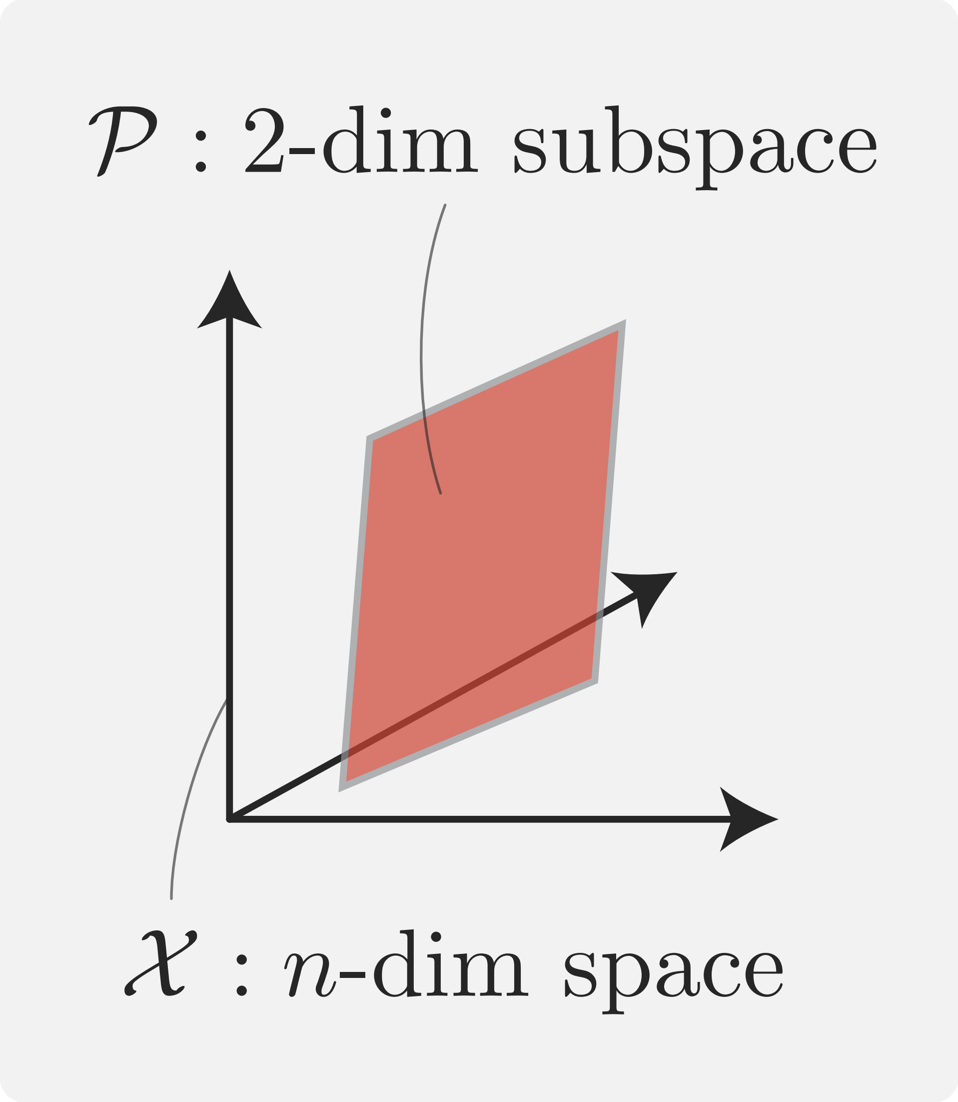
Algorithms to be Compared

Baseline 1:
SLS



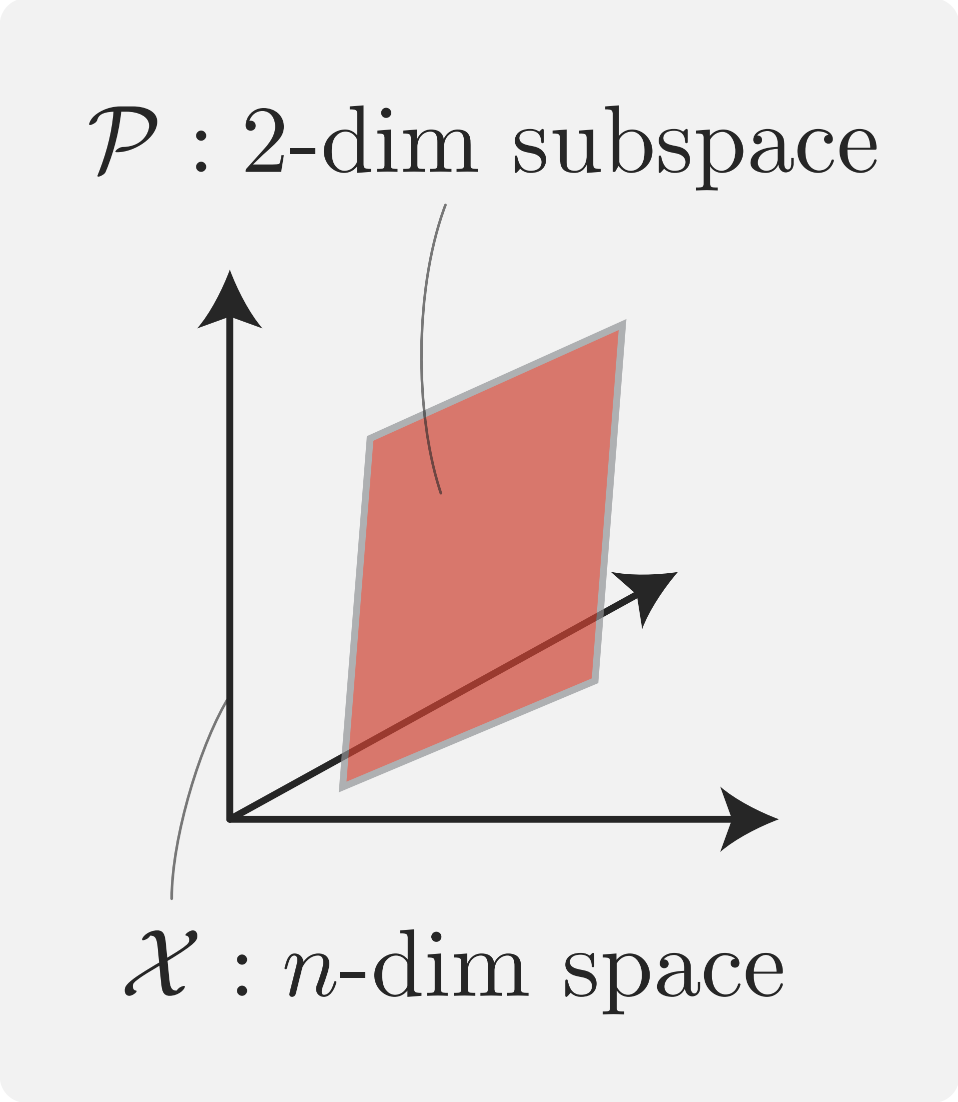
Sequential line search [Koyama+17]

Baseline 2:
SPS (Random)



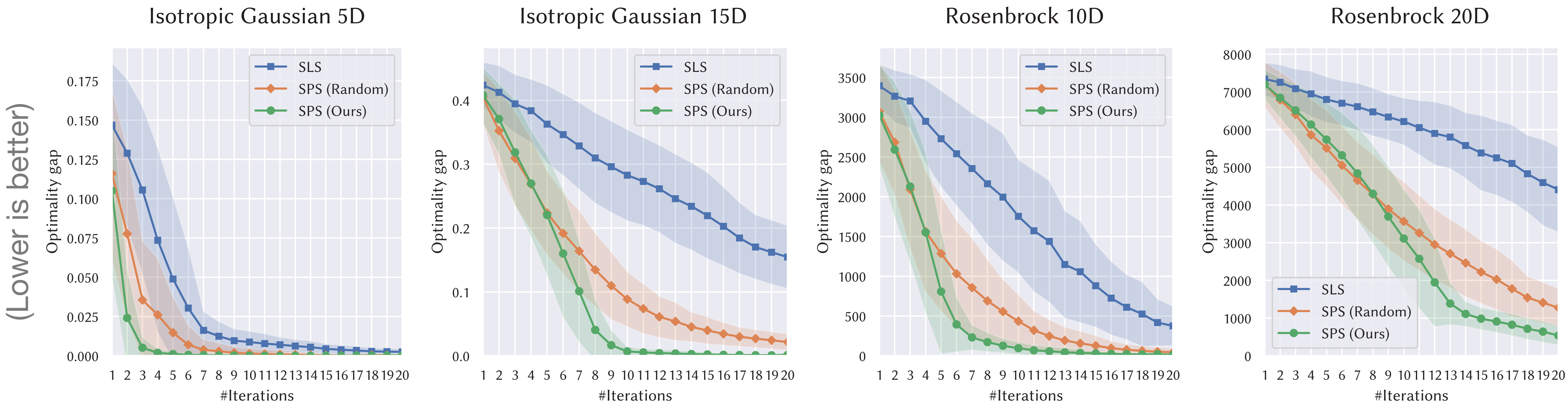
Sequential plane search, but the plane is randomly chosen (instead of using the acquisition function)

SPS (Ours)



Sequential plane search (using the acquisition function)

Result: Performance Comparison



Performance: SLS < SPS (random) < SPS (ours)

$$\max_{\mathbf{x}} \ \mathrm{Preference}(\mathbf{x})$$

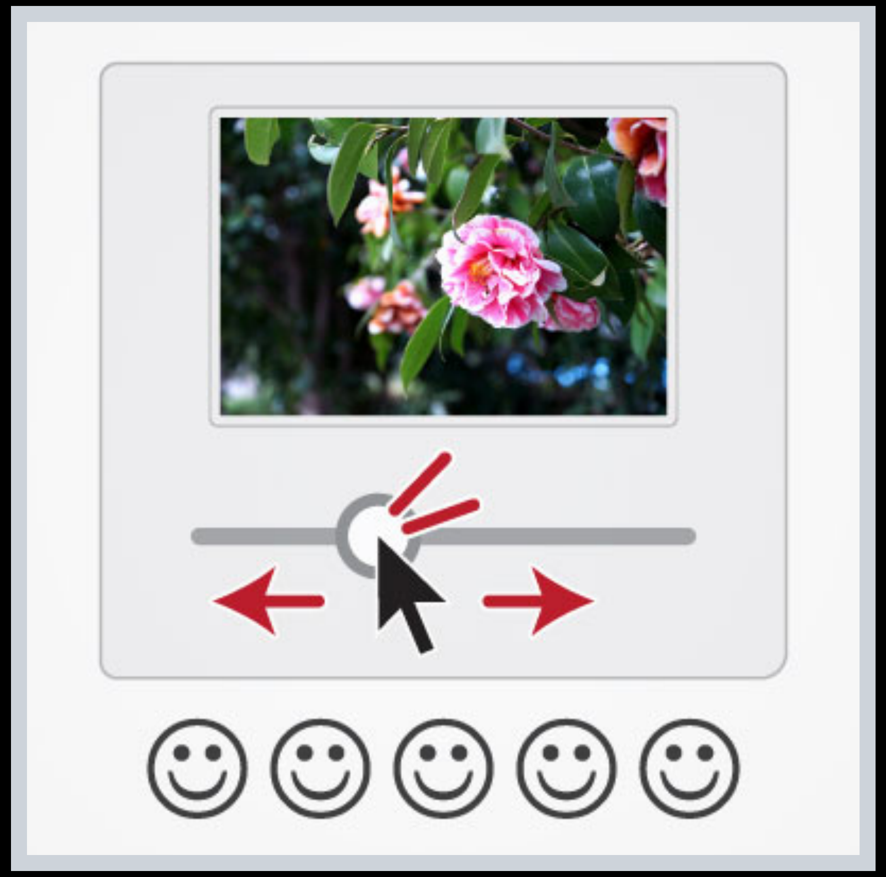
User-in-the-loop Bayesian optimization with **a tailored interface** could solve this problem with only a small number of iterations

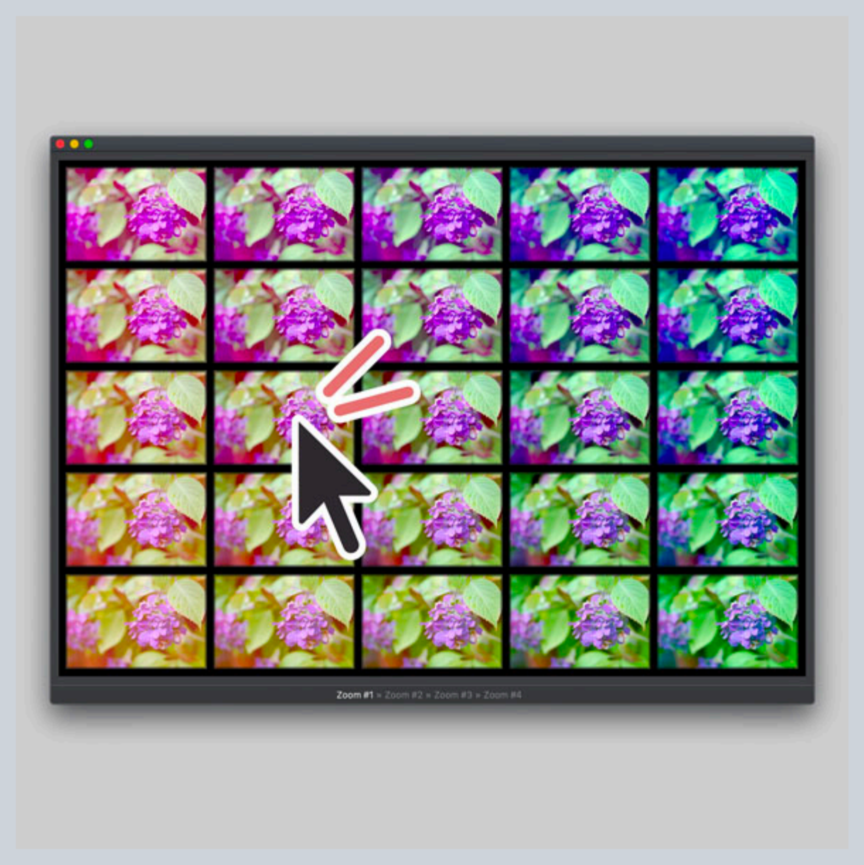
$$\max_{\mathbf{x}} \ \mathrm{Preference}(\mathbf{x})$$

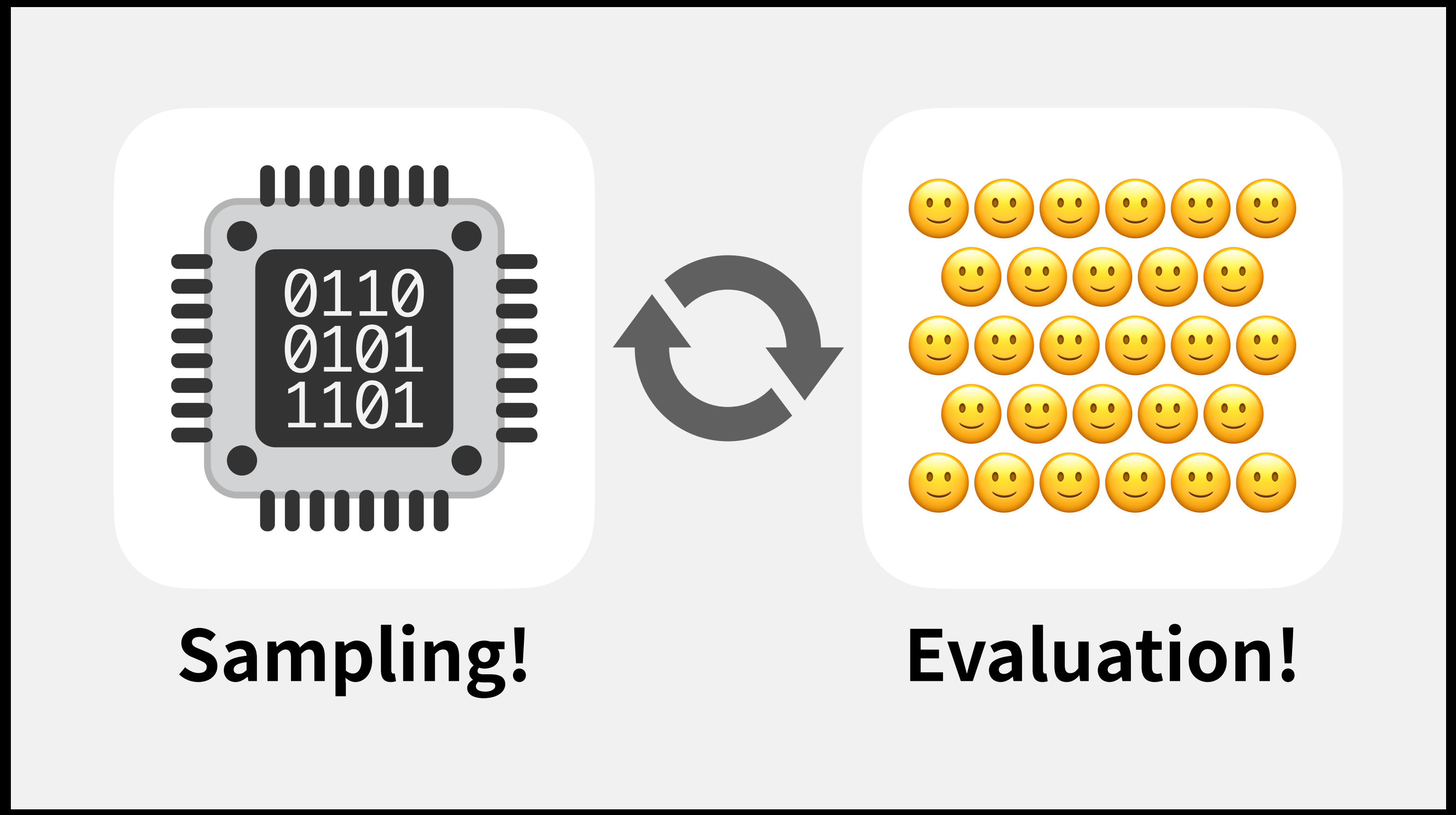
User-in-the-loop Bayesian optimization with **a tailored interface** could solve this problem with only a small number of iterations

⬆

This efficiency was achieved by the tight integration between algorithm design and interface design

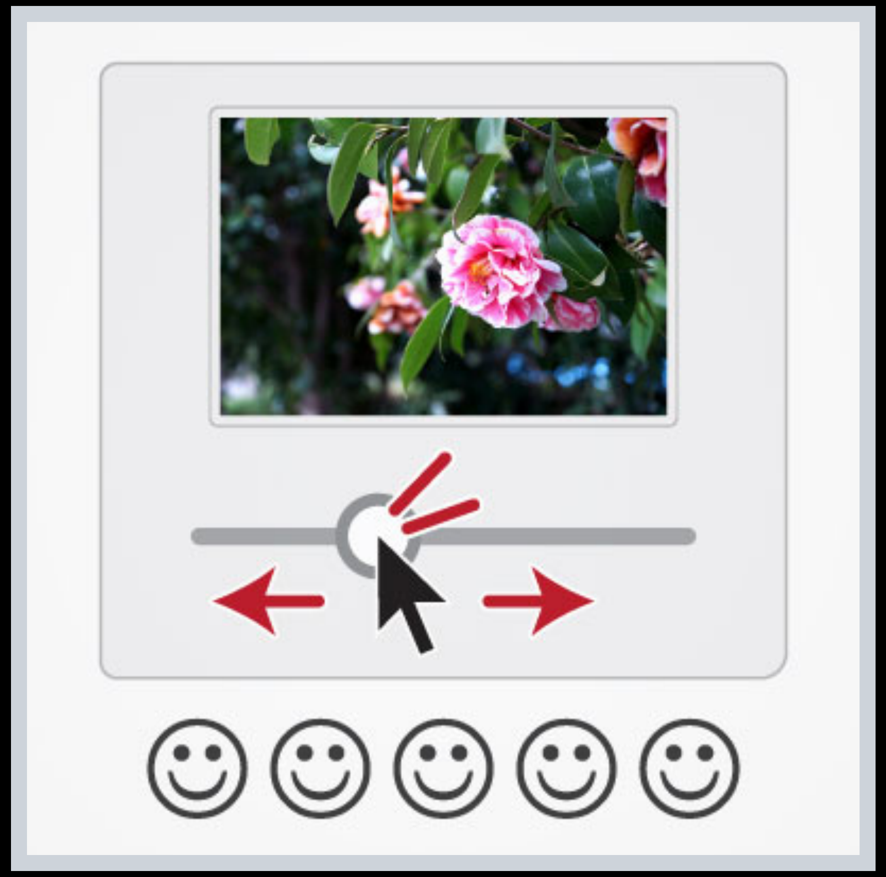


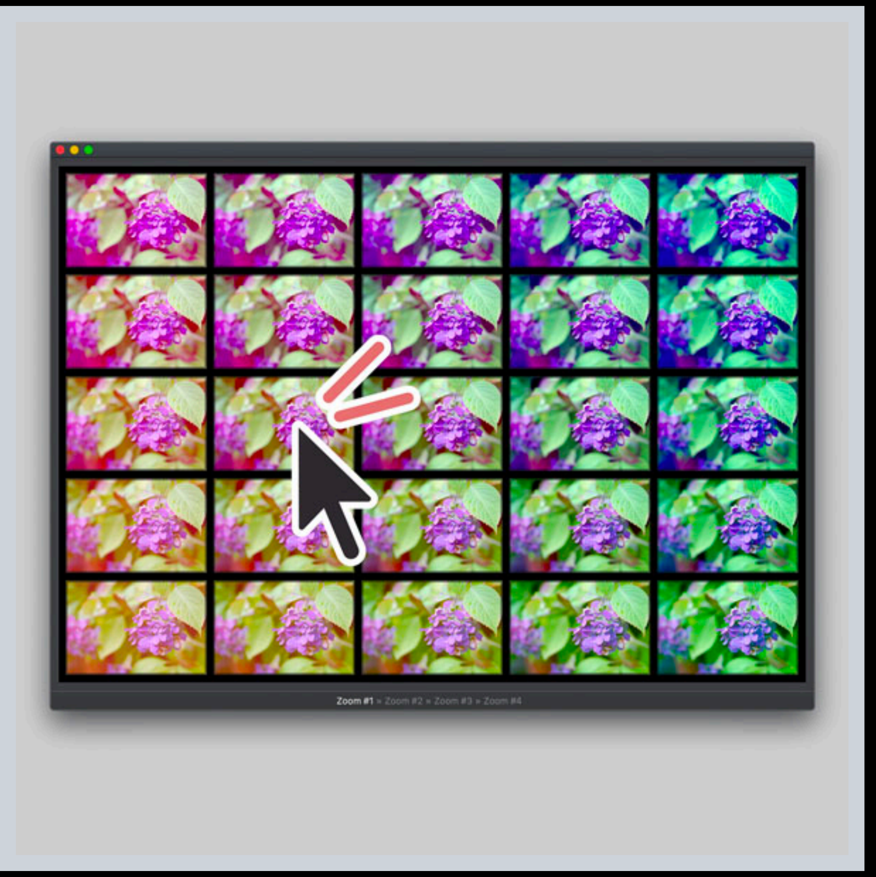


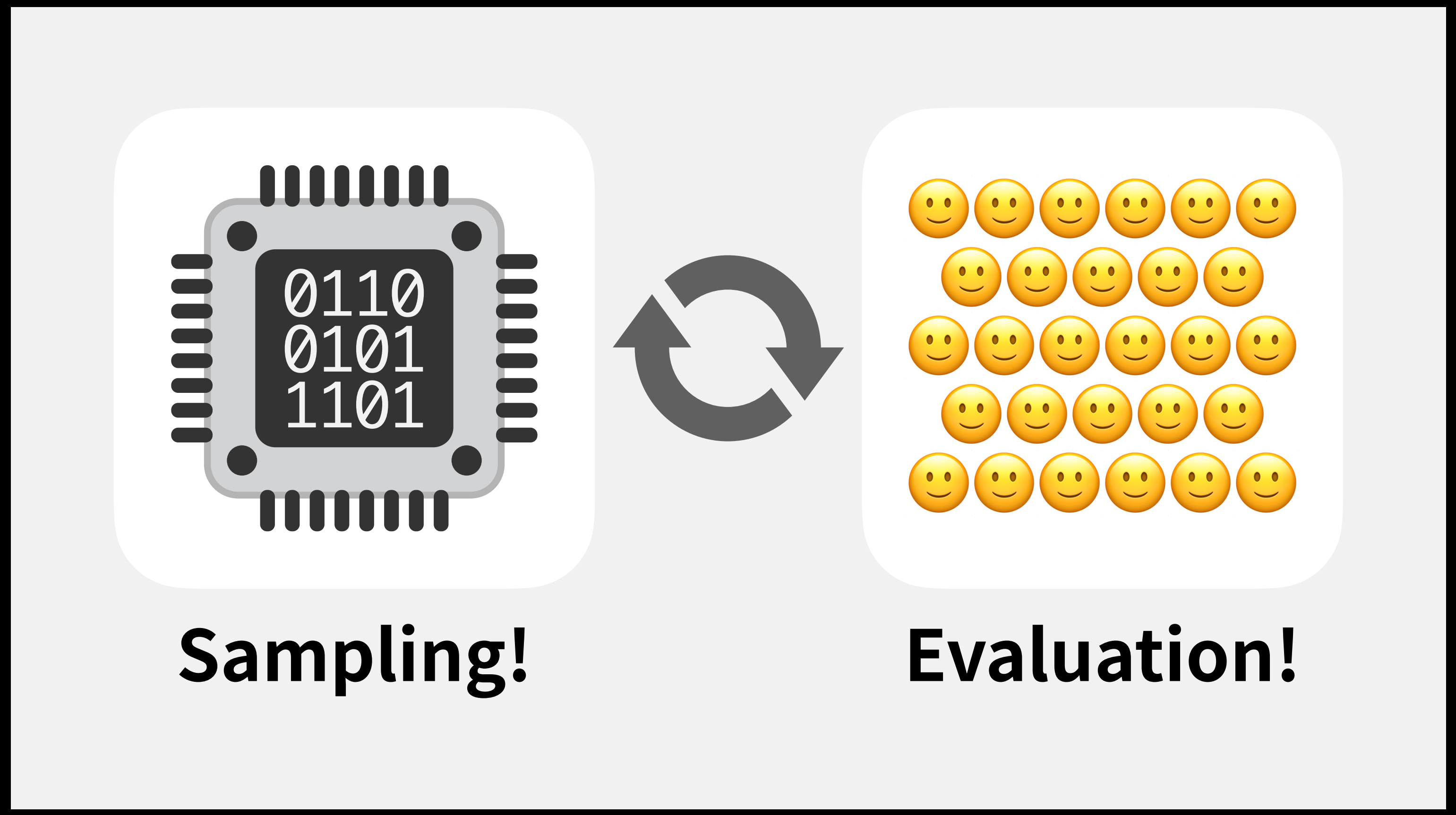


The objective function is defined **solely by human preference**

Loop: the system provides candidates, and the evaluators evaluate them





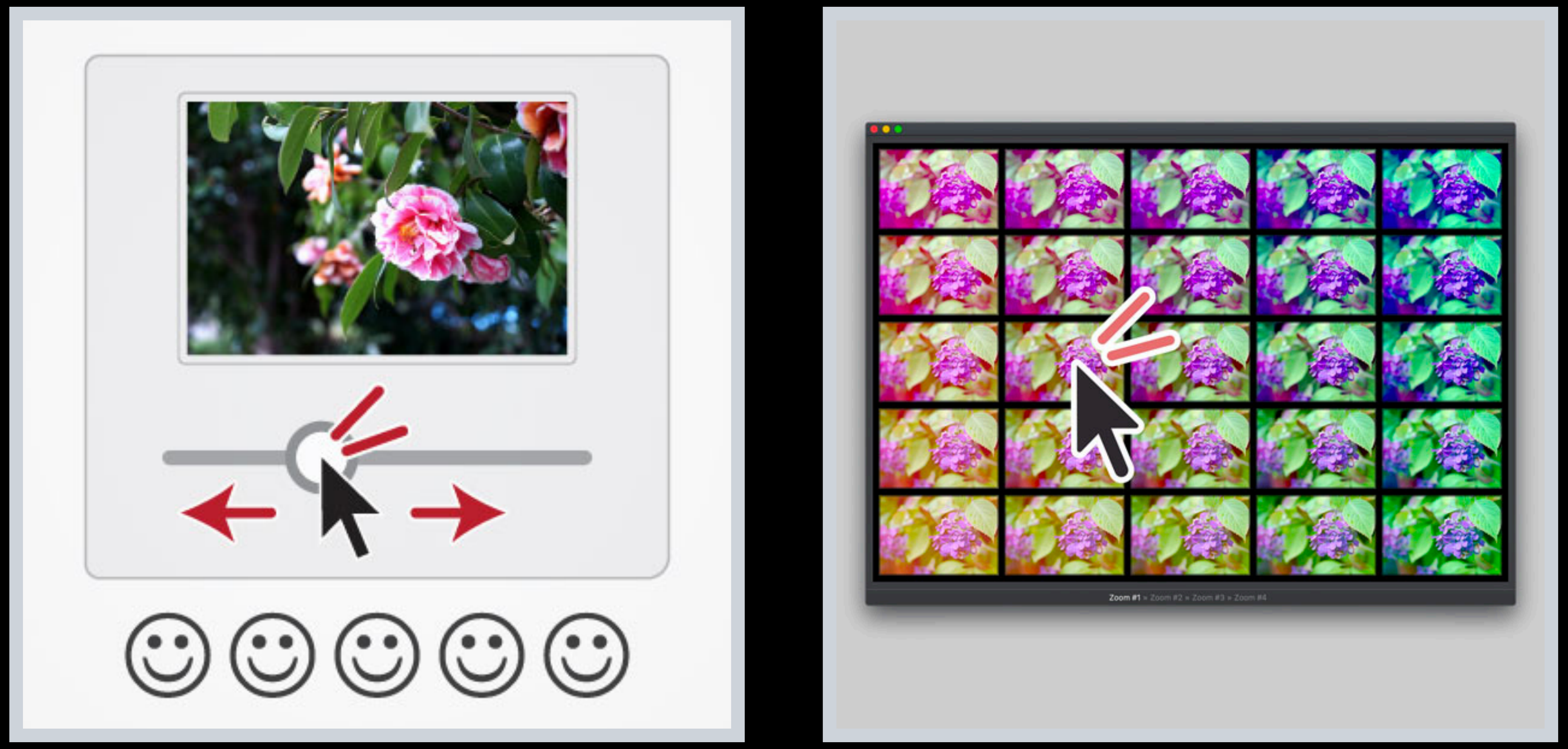


The objective function is defined **solely by human preference**

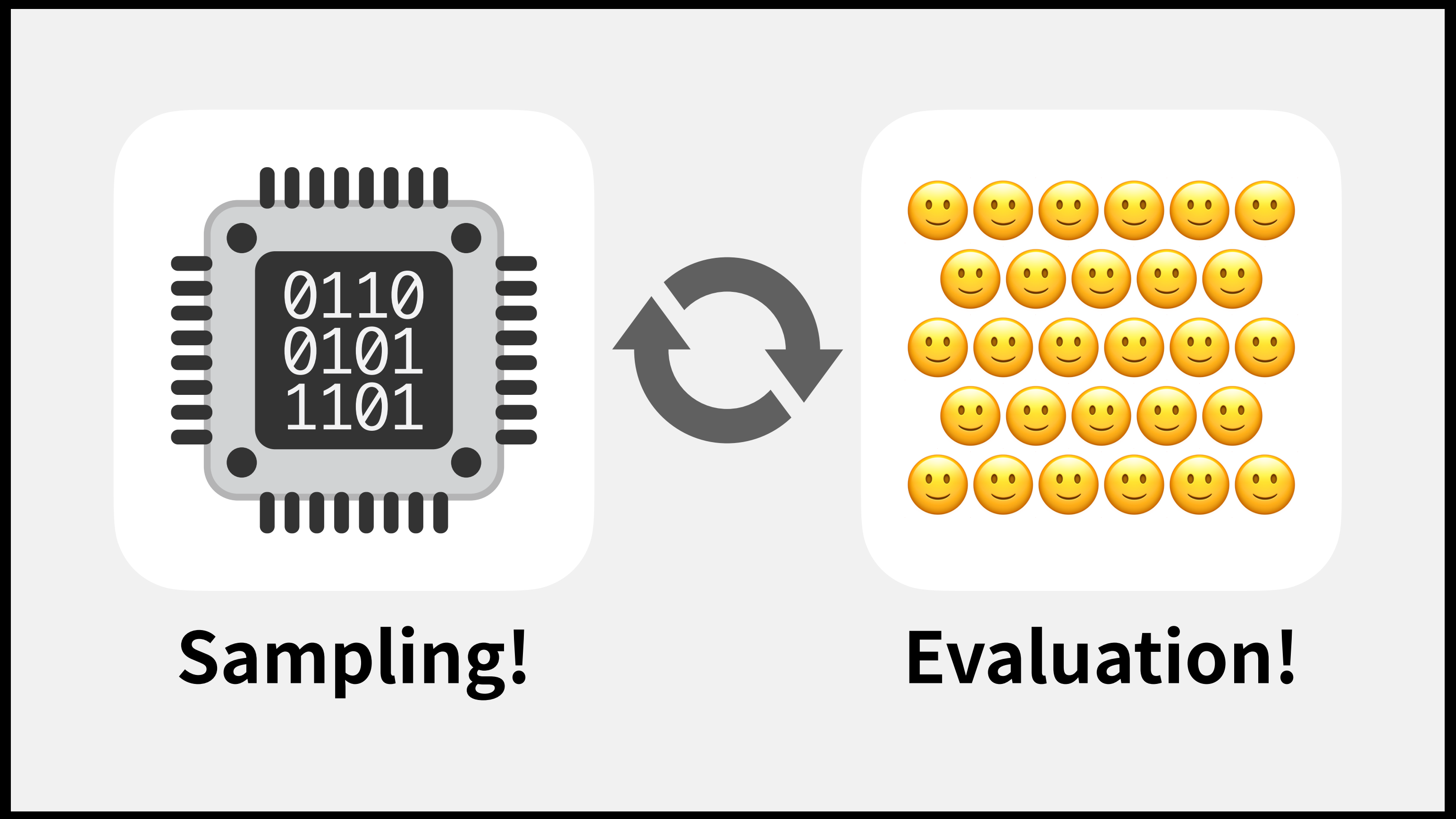
Loop: the system provides candidates, and the evaluators evaluate them

---

The objective function is mostly defined in a machine-executable way, but **needs to be adjusted by human preference**



The objective function is defined **solely by human preference**



Loop: the system provides candidates, and the evaluators evaluate them

---

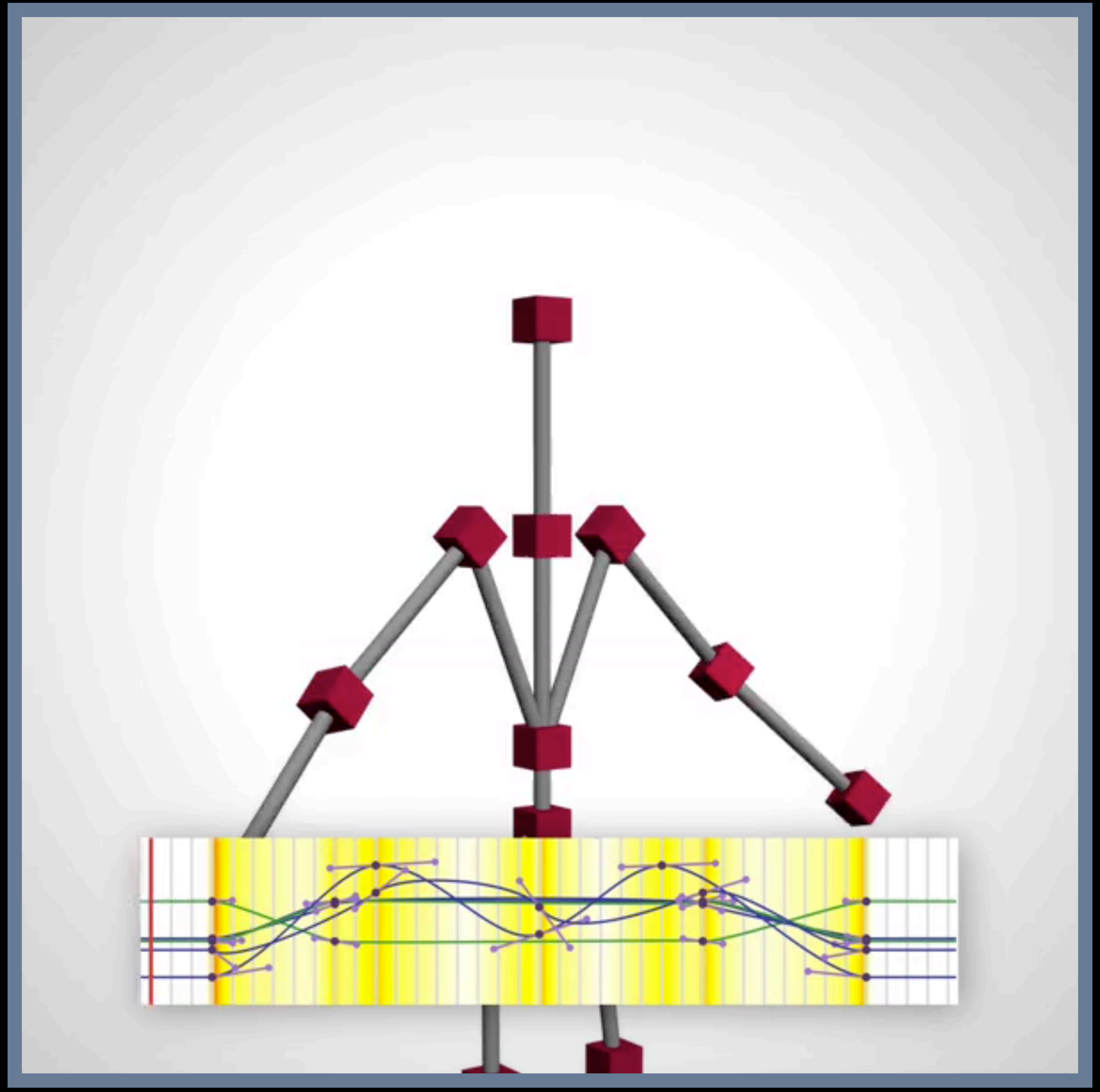
The objective function is mostly defined in a machine-executable way, but **needs to be adjusted by human preference**

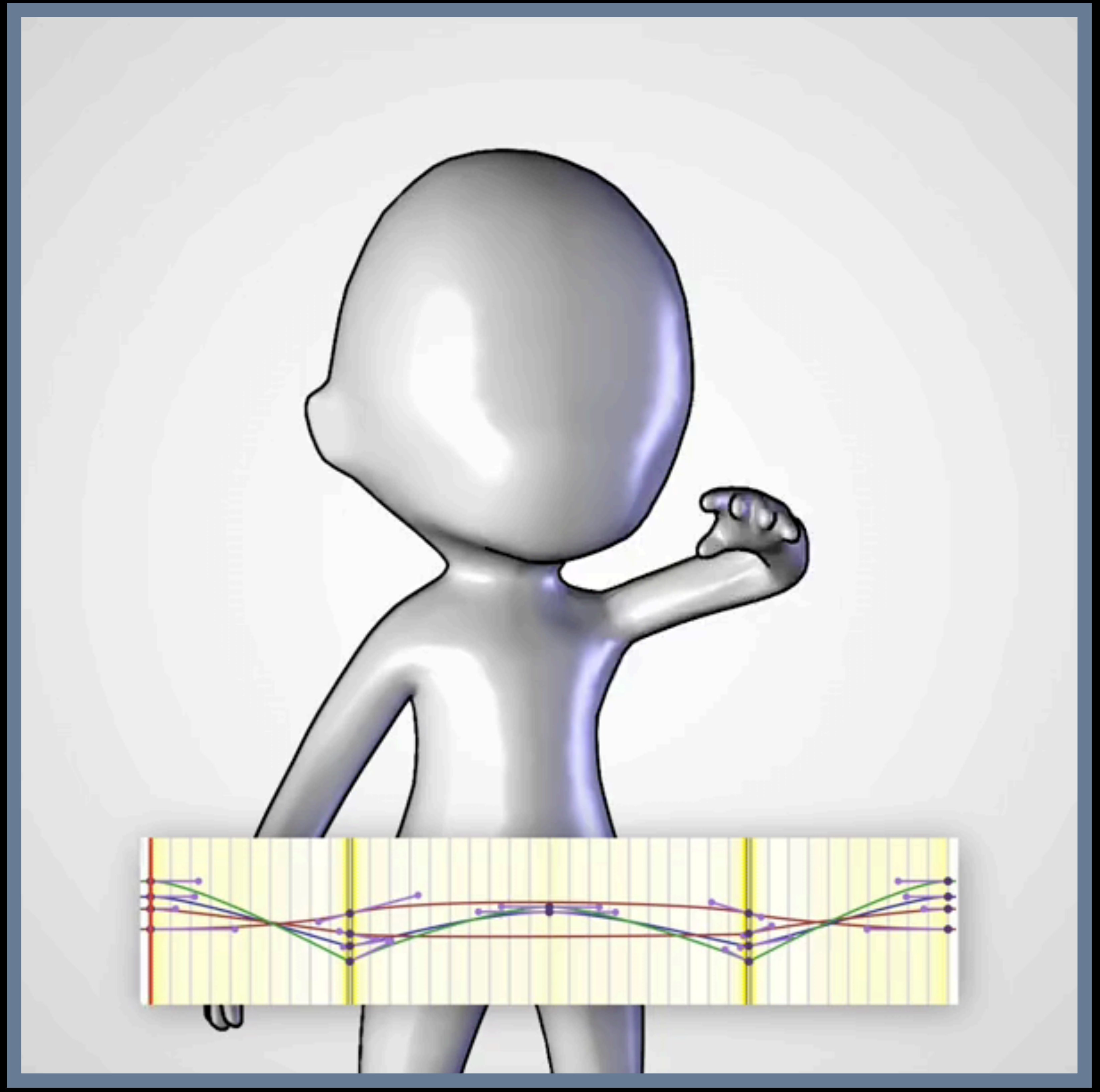
How can the system involve human in the loop?

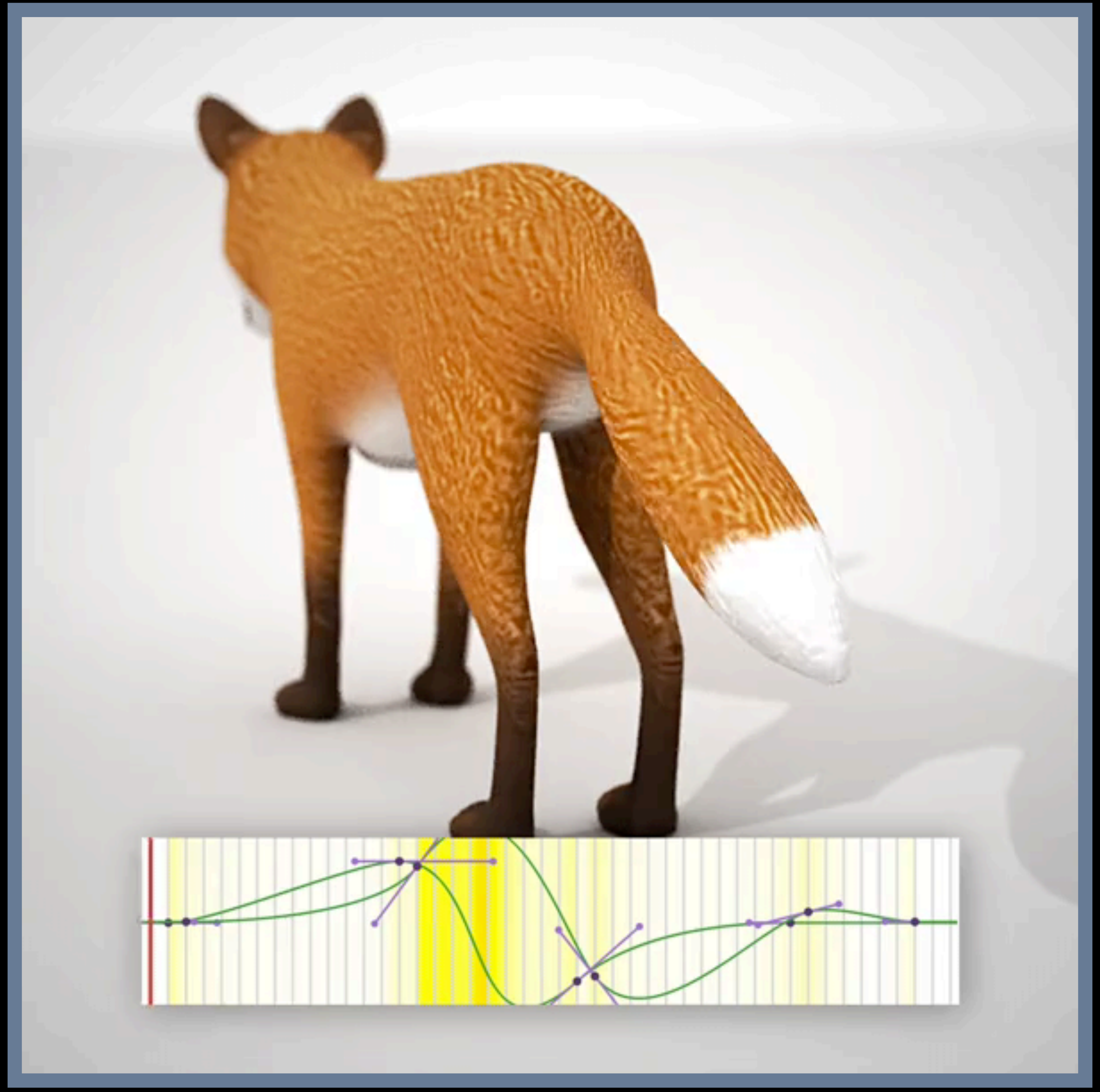
OptiMo: Optimization-Guided Motion Editing for Keyframe Character Animation

Y. Koyama and M. Goto

CHI 2018



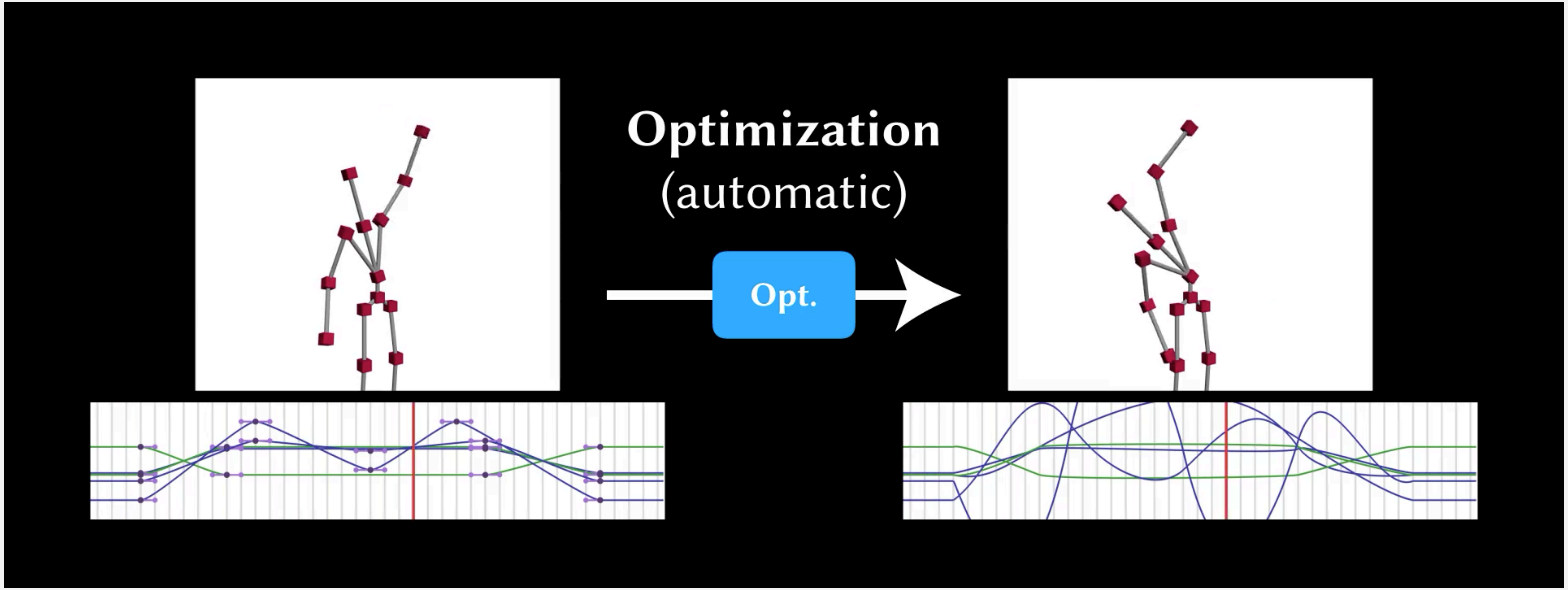




How can animators effectively utilize

optimization techniques in motion editing?

Optimization for Full Automation?



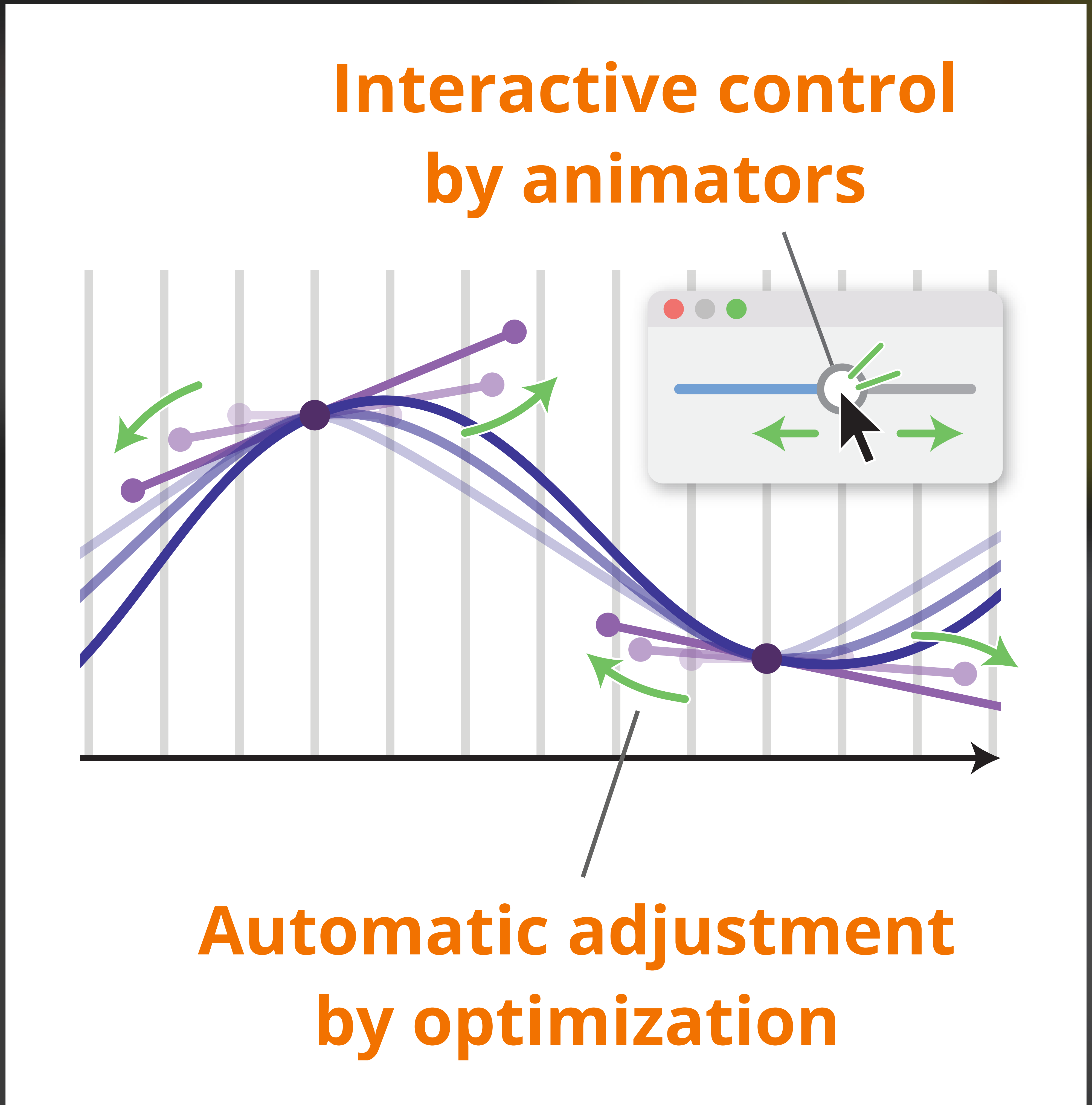
Optimization can fully automate the task,
but results are **not always satisfactory (no control…)**

We support real-time manual control over automatic adjustment by optimization

$$\min_{\mathbf{x}} \mathrm{Cost}(\mathbf{x}, \theta)$$

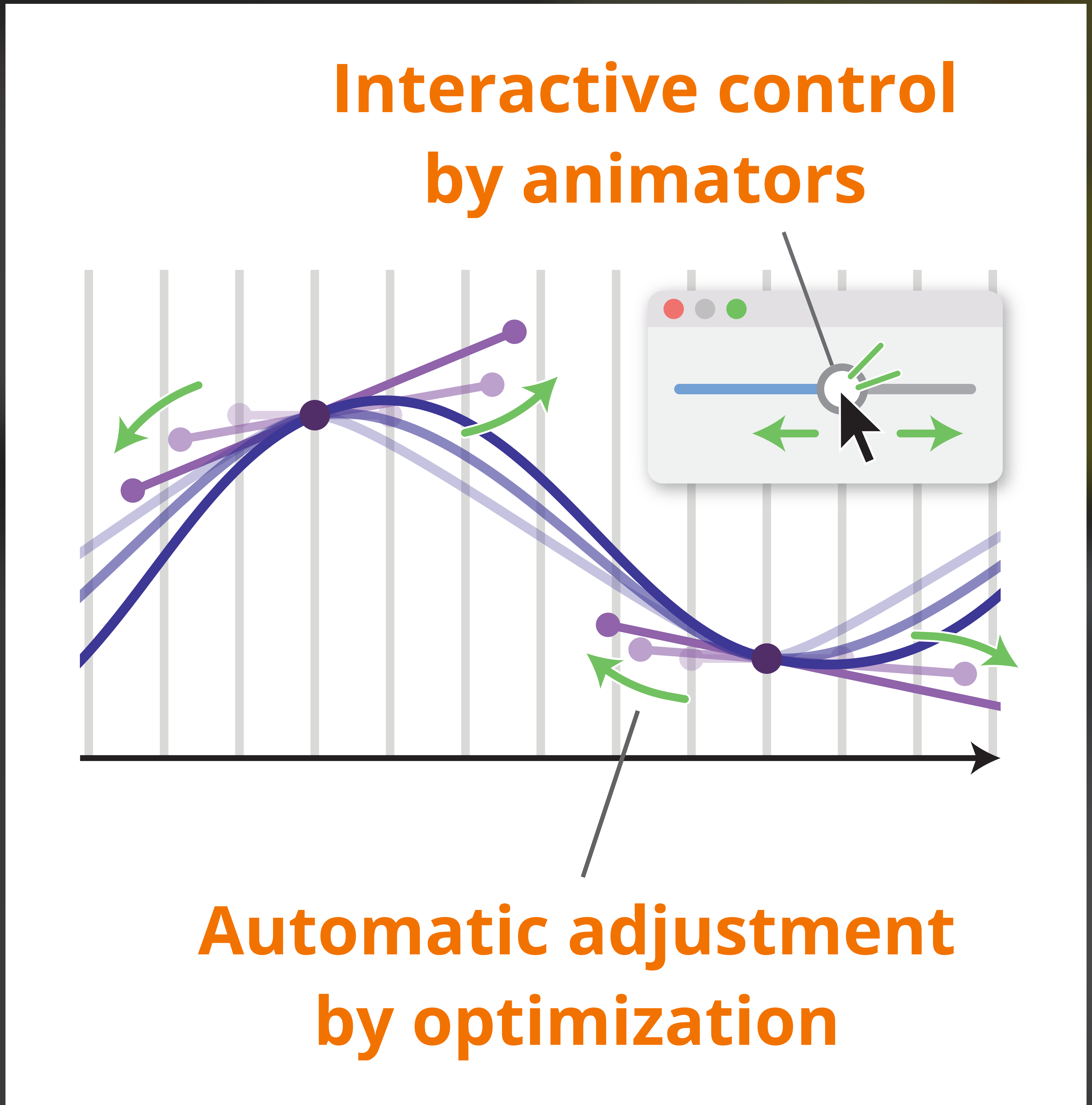
Search variables:
Interpolation curve parameters
for keyframe animation

Objective function:
Motion smoothness, which has
been used in graphics research



$$\min_{\mathbf{x}} \mathrm{Cost}(\mathbf{x}, \underline{\theta})$$

Hyperparameters to control the objective function by adjustment **“during”** the optimization loop



$$\min_{\mathbf{x}} \mathrm{Cost}(\mathbf{x}, \underline{\theta})$$

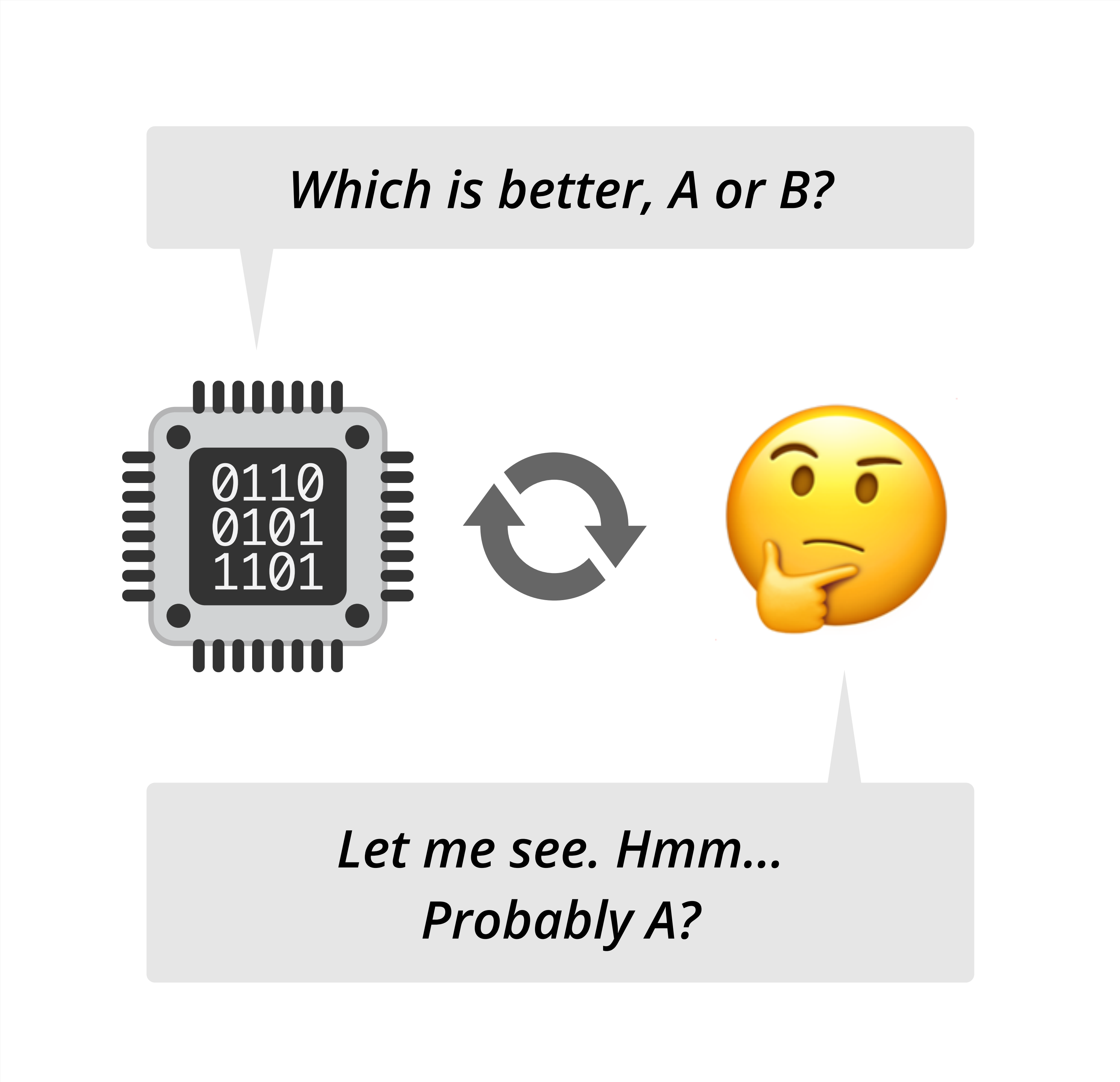
Hyperparameters to control the objective function by adjustment **"during"** the optimization loop

➡ **Another approach to human-in-the-loop optimization, where the objective function is adjustable by the user**

Part 2: Summary

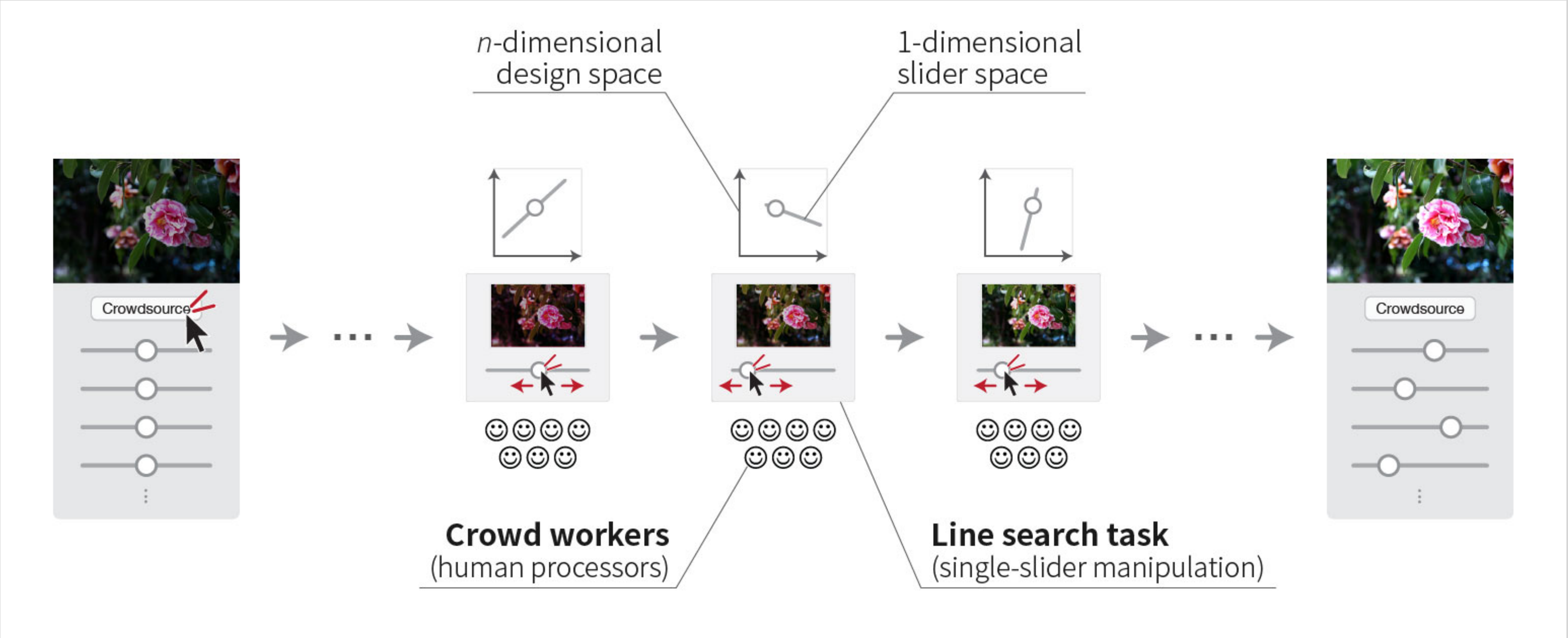
Summary [1/2]

- **Human-in-the-loop optimization** can be used for solving problems with **perceptual objective functions** (e.g., preference)
- **Preferential Bayesian optimization (PBO)** is a promising technique
  - PBO can **reduce the number of necessary queries** (c.f., human is expensive-to-evaluate)
  - PBO can use **relative assessment** (c.f., human is not good at providing absolute assessment)

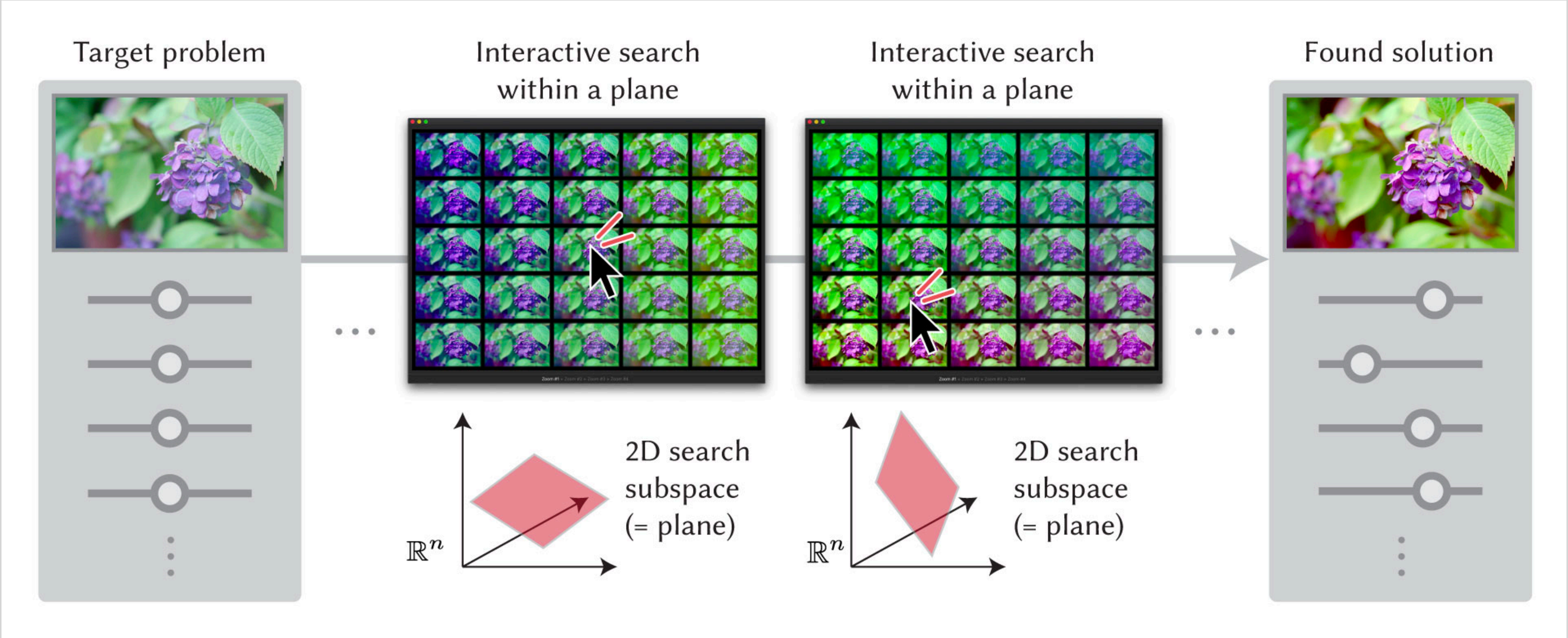


Summary [2/2]

- **Tight integration between algorithm design and interface design** is the key to achieve **higher efficiency** in human-in-the-loop optimization systems



Sequential line search
[SIGGRAPH 2017]



Sequential plane search
[SIGGRAPH 2020]

Introduction to Computational Design

Part 2: Human in the Loop

Presenter: Yuki Koyama (AIST, Japan)

2021-05-12 | CHI 2021 Course | Online