LightGuider: Guiding Interactive Lighting Design using Suggestions, Provenance, and Quality Visualization


https://www.computer.org/csdl/journal/tg/5555/01/08807288/1cG6djufx96
Lighting design

The process of placing light such that the emitting light fulfills technical and aesthetic requirements.
Lighting design is complex

- Must satisfy design constraints
- Must look good
- Simulating lighting is computationally expensive
  - Select, place, and align lights → run simulation
  - Check if illumination constraints are satisfied
  - Repeat until all constraints are fulfilled and design looks good
- Designers generally converge on solutions: single local optimum
LightGuider

Simulates potential next modeling steps and shows how well current designs meet specified quality criteria.
LightGuider

(a) 3D modelling view
LightGuider

(a) 3D modelling view

(b) Provenance tree view
LightGuider

(a) 3D modelling view

(b) Provenance tree view

(c) Screenshots view
LightGuider

(a) 3D modelling view

(b) Provenance tree view

(c) Screenshots view

(d) Quality view
LightGuider

(a) 3D modelling view
(b) Provenance tree view
(c) Screenshots view
(d) Quality view
(e) Focus setting view
Video

https://vimeo.com/360154391
Components of LightGuider

- 3D modelling view
- Quality view
- Provenance tree view
  - Displays design suggestions
- Focus setting view
- Screenshots view
3D Modelling View

● LightGuider is built on top of a lighting design tool

Specific to LightGuider:

● Camera animations towards poorly performing objects
● Displays colored outlines around selected objects
Quality View

- Shows all illumination constraints and current status
- Positions on aligned but not common scales
- Hue maps to different constraints
- Dark, saturated → solution is far off
- Light → solution is close
- Scales have equal brightness values on all levels
Provenance Tree View

- Node-link diagram that shows workflow history
- Letters indicate different actions
- Select a node to highlight path towards it
Provenance Tree View

- Treemap in each node
- Each constraint associated with distinct color
  - Same as quality view
Provenance Tree View

- Can show more details on demand
Provenance Tree View

- Can compare modelling states globally
- Grayscale
- Selected node acts as reference; all other nodes encode the difference to it
- Darker → worse
- Lighter → better
Focus Setting View

- Set weights for illumination constraints and user-defined groups
- Slider colors match the colors in the tree nodes
- More weight → larger corresponding area in tree node
- More weight → more important when generating design suggestions
Screenshots View

- Shows thumbnails for linear path through tree to current state
- Thumbnails also shown at leaf nodes
Provenance Tree View: Analysis

- Node-link diagram
  - Horizontal growth shows development through time
- Treemap summarizes modelling step
  - Color encodes constraint status
  - Area encodes constraint weight
  - Spatial position does not encode data
- Difficult to read hierarchy info from treemap, but not important in this scenario
Generating lighting design suggestions

- Can add, remove, dim, or change lights
- Can change height of one or all lights
Generating lighting design suggestions

- Can add, remove, dim, or change lights
- Can change height of one or all lights

Process:

- Compute scores for all actions, accounting for weights assigned to illumination constraints
- Pick top 2 actions, simulate 3-5 randomized parameterizations for each
- Compute scores for random simulations, accounting for weights assigned to illumination constraints
- Show 3 highest-scoring solutions to user
Summary

● What
  ○ Workflow history (network), design quality (quantitative values)

● Why
  ○ Generate and verify satisfactory designs
  ○ Discover alternate design paths

● How
  ○ **Encode:** node-link diagram, treemap, horizontal scales
  ○ **Manipulate/facet:** update scene, select step to compare it to all other steps
  ○ **Reduce:** aggregate constraint statuses
Overall Critique
Strengths

- Justifies design choices for specific tasks
- Implements overview then details on demand
- Follows “eyes beat memory”
- Recognizes limitations in scalability
  - Hues
  - Nodes in provenance tree
Weaknesses and limitations

- Examples of scalability of provenance tree
- Justification for randomly generated suggestions
  - Came up in user study feedback
- Clarity of LightGuider’s 3D modelling view contribution
Thank you