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

Andreas Walch, Michael Schwäzte, Christian Lukas, Elmar Eisemann, Theresa Gschwandtner. TVCG (Proc. VAST/CHI/Vis/SoVW 2019 Special Issue)

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 assign 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

Components of LightGuider

- 3D Modelling View
  - Specific to LightGuider:
  - Focus setting view
- Quality View
  - Shows all illumination constraints and current status
  - Positions on aligned but not common scales
  - High 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
  - Nodes indicate different actions
  - Select a node to highlight path towards it

LightGuider

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

Video

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

- Can show more details on demand
Provenance Tree View
- Can compare modeling 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

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

Strengths
- Justifies design choices for specific tasks
- Implements overview then details on demand
- Follows “eyes beat memory”
- Recognizes limitations in scalability
  - Colors
  - 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 LightGlider’s 3D modeling view contribution

Overall Critique

Thank you