TextureFusion
High-quality Texture Acquisition for Real-time RGB-D Scanning

CVPR 2020 Oral

Joo Ho Lee†  Ha Hyunho†  Yue Dong§  Xin Tong§  Min H. Kim†

† KAIST  § MSRA
High-quality texture acquisition for real-time RGB-D scanning

Voxel representation

TextureFusion (ours)
Per-voxel color representation

Color per voxel*

Higher resolution → lower performance

Blurriness of texture

Traditional texture mapping

• Two-fold reconstruction process

• Global optimization of multiple views

• **Real-time computation impossible!**

Patchmatch based optimization
[Bi et al. 2017]

Color optimization
[Zhou and Koltun 2014]
Key insight

• Keep 2D topology of texture without meshing in REAL-TIME!
Tile-based texture data structure

```
struct texel {
    uchar color[3];
    uchar weight;
}
```
Real-time texture integration framework

Input

Depth map $D_t$  Image $I_t$

Previous

Geometry $S_{t-1}$  Texture $Q_{t-1}$

Frame integration

Current

Geometry $S_t$  Texture $Q_t$
Real-time texture integration framework

$S_{t-1}$

$Q_{t-1}$
Real-time texture integration framework

Previous

geometry $S_{t-1}$

texture $Q_{t-1}$

Depthmap $D_t$

Surface integration

Current

geometry $S_t$
Real-time texture integration framework

Previous

geometry $S_{t-1}$

texture $Q_{t-1}$

Depthmap $D_t$

Surface integration

Current

geometry $S_t$

Not valid
Real-time texture integration framework

Surface integration

Geometric correspondence

Texture transfer

Depthmap $D_t$

Previous

geometry $S_{t-1}$
texture $Q_{t-1}$

Current

geometry $S_t$
texture $Q_t$
Real-time texture integration framework

- Previous
  - Geometry $S_{t-1}$
  - Texture $Q_{t-1}$

- Current
  - Geometry $S_t$
  - Texture $Q_t$

- Depthmap $D_t$
- Surface integration
- Geometric correspondence
- Texture transfer
- Texture transfer $\tilde{Q}_t$
- Valid
Texture-image misalignment

- Camera pose estimation using the geometric information
- Not guarantee the photometric consistency of the integrated texture map

Image projection without warp
Texture-image correspondence search

- Estimate a spatially-varying camera motion field

Current pose

New pose

3D surface point

Camera motion grid
Spatially-varying perspective correction

- Maximize the photometric consistency of local 3D surface patches in the current texture map
Hierarchical optimization of motion field

0-level motion estimation

2-level motion grid \( \{ \tau_i^0 \} \)

Initialization

1-level motion estimation

1-level motion grid \( \{ \tau_i^1 \} \)

Initialization

2-level motion estimation

0-level motion grid \( \{ \tau_i^2 \} \)
Real-time texture integration framework

- Previous
  - Geometry $S_{t-1}$
  - Texture $Q_{t-1}$
- Depthmap $D_t$
- Surface integration
- Texture transfer
- Current
  - Geometry $S_t$
  - Texture $Q_t$
- Image $I_t$
- Texture warp & integration
Real-time RGB-D scanning with our texture fusion
Conclusions

Texture reconstruction framework

Tile-based texture data structure

Code available: https://github.com/KAIST-VCLAB/texturefusion

Narrated by Mustafa Yaldiz