Implicit Neural Representations: From Objects to 3D Scenes

Andreas Geiger

Autonomous Vision Group
University of Tübingen / MPI for Intelligent Systems Tübingen

June 19, 2020
Collaborators

Songyou Peng
Michael Oechsle
Carolin Schmitt
Michael Niemeyer
Lars Mescheder
Simon Donne
Gernot Riegler
Vladlen Koltun
Marc Pollefeys
Andreas Geiger
3D Representations

- Traditional Explicit Representations ⇒ Discrete
- Implicit Neural Representation ⇒ Continuous

Limitations

Structure of implicit neural representations:

- Global latent code $\Rightarrow$ no local information, overly smooth geometry
- Fully connected architecture $\Rightarrow$ does not exploit translation equivariance

Limitations

Implicit models work well for simple objects but **poorly on complex scenes:**
How to reconstruct large-scale 3D scenes with implicit neural representations?

Convolutional Occupancy Networks
Convolutional Occupancy Networks

- **2D Plane Encoder**: Local PointNet processes input, project onto canonical plane
- **2D Plane Decoder**: Processed by U-Net, query features via bilinear interpolation
- **Occupancy Readout**: Shallow occupancy network $f_\theta(\cdot)$

Convolutional Occupancy Networks

▶ **3D Volume Encoder:** Local PointNet processes input, volumetric feature encoding

▶ **3D Volume Decoder:** Processed by 3D U-Net, query features via trilinear interp.

▶ **Occupancy Readout:** Shallow occupancy network $f_\theta(\cdot)$
Comparison

Occupancy Networks

Convolutional Occupancy Networks

Results
Object-Level Reconstruction

Input  ONet  Ours  GT

Training Speed

![Graph showing training speed and validation IoU over training iterations](image)

- ONet
- PointConv
- Ours-2D (64²)
- Ours-2D (3 × 64²)
- Ours-3D (32³)

Training Speed

![Training Speed Graph](image)

Scene-Level Reconstruction

▶ Trained and evaluated on synthetic rooms

Scene-Level Reconstruction

▶ Trained on synthetic rooms, evaluated on ScanNet

Results on Matterport3D

- Fully convolutional model
- Trained on synthetic crops
- Sliding window evaluation
- Scales to any scene size
Key Insights:

- Convolutional models allow for scaling implicit models to larger scenes
- Convolutional models train faster than fully implicit models
- Convolutional models allow for incorporating local feature information
- For objects, the 3-plane model has the best accuracy/memory trade-off
- For scenes, the volumetric representation performs best
- Models transfer from synthetic to real scenes
How to capturing the visual appearance of objects?

Conditional Surface Light Fields
Problem Definition

Existing Representation

**Texture Fields**

- 3D consistent
- Generalize across objects
- View-point independent
- Do not model lighting

[Oechsle et al., ICCV 2019]
Conditional Surface Light Field

Rendering equation:

\[ L(p, v, l, n) = \int_{\Omega} s v \text{BRDF}(p, r, v) \cdot l(r) \cdot (n^T r) \, dr \]

Conditional surface light field:

\[ L_{\text{CSLF}}(p, v, l) : \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}^M \to \mathbb{R}^3 \]

Overfitting to Single Objects

Conditional Implicit Surface Light Field (cSLF)

Appearance Field

\[ a_\theta : \mathbb{R}^3 \times S \times Z \rightarrow \mathbb{R}^D \]

Lighting Model

\[ l_\theta : \mathbb{R}^D \times \mathbb{R}^2 \times \mathbb{R}^M \times S \rightarrow \mathbb{R}^3 \]

Light setting

3D Point

Input Shape

View direction

Loss

Predicted Image

Target Image

Single-Image Appearance Prediction

Generative Model

How to obtain training data with materials?

Joint Estimation of Pose, Geometry and svBRDF
Joint Estimation of Pose, Geometry and svBRDF

**Goal: Dataset of 3D indoor scenes**
captured with high accuracy from a handheld mobile sensor.

**Custom built sensor rig:**
- Custom IR depth sensor similar to Microsoft Kinect
- Active illumination + RGB camera for material estimation
Joint Estimation of Pose, Geometry and svBRDF

Materials $\leftrightarrow$ Geometry

$\rightarrow$ Accurate geometry reconstruction requires known appearance properties

$\leftarrow$ Accurate appearance estimation requires very well known geometry

$\leftrightarrow$ Joint estimation requires only a rough initialization for both

Joint Estimation of Pose, Geometry and svBRDF

Contributions:

- **Joint** formulation
- **Single objective function**
  minimized using off-the-shelf gradient-based solvers
- **Meaningful segmentation**
  differentiably part of the optimization
- **Accurate geometry**
  with very fine details

\[
\mathbf{x}^* = \arg\min_{\mathbf{x}} \mathcal{L}(\mathbf{x})
\]
Joint Estimation of Pose, Geometry and svBRDF

Contributions:

▶ Joint formulation
▶ Single objective function
  minimized using off-the-shelf gradient-based solvers
▶ Meaningful segmentation
  differentiably part of the optimization
▶ Accurate geometry
  with very fine details

Reconstruction    Segmentation

Joint Estimation of Pose, Geometry and svBRDF

Contributions:

► **Joint** formulation

► **Single objective function**
  minimized using off-the-shelf gradient-based solvers

► **Meaningful segmentation**
  differentiably part of the optimization

► **Accurate geometry**
  with very fine details

Qualitative Results

Conclusion:

- Joint estimation helps
- This is only a first step
- Object-level reconstruction remains challenging with limited observations
- Scaling to larger scenes
- Scaling to scenes with external illumination
How to obtain training data with semantic labels?

KITTI-360
KITTI-360

Sensors:

- Front-facing stereo camera
- 360° fisheye cameras
- Velodyne HDL 64 laser scanner
- SICK pushbroom laser scanner
- IMU/GPS localization system

Features:

- Driving distance: **73.7 km**  Frames: **4 × 83,000**
- All frames accurately **geolocalized** (⇒ OpenStreetMap)
- Semantic label definition consistent with Cityscapes, **19 classes** for evaluation
- Each instance assigned with a **consistent instance ID** across all frames

Xie, Kiefel, Sun, Geiger: Semantic Instance Annotation of Street Scenes by 3D to 2D Label Transfer. CVPR, 2016
Sensors

Wheel axis (0.30 m)

GPS/IMU (0.9 m)

Cam 1 2.71 m 0.81 m 0.32 m 0.79 m 0.60 m 0.05 m 0.48 m

Cam 2

Cam 3 0.92 m 0.08 m

Cam 4

Fisheye cameras (1.95 m)

Perspective cameras (1.55 m)

Velodyne (1.73 m)

SICK (1.69 m)

GPS/IMU (0.9 m)

All heights wrt. road surface

Camera inclination: ~5° (down)
360° 2D Sensors
360° 3D Sensors

- Velodyne
- SICK
- Stereo
3D Annotations

RGB

Bounding Box

Semantic

Instance
2D Annotations

Semantic

Instance

Confidence

Bounding Box
Thank you!

http://autonomousvision.github.io