

Quelle: <https://aps.arxiv.org/pdf/2009.09808v1.pdf>

Neural implicits (Neural networks)

Paper

- **Overfit Neural Networks as a Compact Shape Representation**
- Von Thomas Davies , Derek Nowrouzezahrai and Alec Jacobson
- In Kanada
- Im Jahr 2020

Agenda

- Einleitung
- Methodik
- Ergebnisse
- Limitierungen & Anwendungen

Einleitung

- SDF
- Problemstellung
- Beitrag

SDF

- Begriff:
eine vielseitige implizite Oberflächendarstellung, die überall in der Computergrafik nützlich ist.
- Darstellung:
Komplexe Objekte können **semianalytisch** erstellt werden durch Zusammensetzen **geometrischen Primitiven mit Raumverzerrung, Überblendungsoperationen und Replikationsfunktionen.**

SDF

➤ Speicherung:

- auf einem regelmäßigen Gitter (Kosten, Profitieren)
- explizite Darstellungen als Datenformat
- Wie wird Mesh-Assets in implizite Darstellungen umwandelt?

SDF

- Konvertieren:
 - Die Überanpassung eines **tiefen neuronalen Netzes** an die SDF effektiv ist, und wir plädieren dafür, es als eine erstklassige implizite Darstellung (Neuronale Implicits).
- Neuronale Implicits
 - Die effektiv unendliche Auflösung von Implicits, aber mit der Berechnungseffizienz von groben Netzen und der Speicherzugriffseinheitlichkeit eines festen Gitters.

Problemstellung

- Die Konvertierung in **Punktwolken** umgeht, dass das Problem der Homogenität, indem die Abhängigkeit von der Ordnung oder der Expliziten / impliziten über die vielfältige Struktur der Form vollständig beseitigt.
- bei Klassifizierungs- und Erkennungsaufgaben
- die Interpolation von Formen, differenzierbares Rendering und Oberflächenrekonstruktion

Problemstellung

“Training a specific neural network for each shape is neither feasible nor very useful.” — PARK, FLORENCE,

Beitrag

- Deep neuronale Netze, oder Neural Implicits, eine Kombination der wünschenswerten Eigenschaften einer Formdarstellung aufweisen. Die an eine einzelne Form wird als **Testfall** behandelt.
- Vergleich die wirtschaftliche Speicherung von Neural Implicits mit bestehenden Formaten

Methodik

- neuronale Netzwerkarchitektur(OVERFITSDF):
 - die darauf trainiert ist, eine Überanpassung an eine einzelne vorzeichenbehaftete Distanzfunktion
 - gelernte Parametersatz als effiziente und leichtgewichtige Darstellung der Form verwendet werden

Methodik

- Neural Implicit Format:
 - die gelernten Netz Gewichte eines OVERFITSDF-Modells, das auf Stichproben aus der vorzeichenbehafteten Distanzfunktion der Form trainiert wurde.

Methodik

- Neural Implicit Dateiformat:
 - einfach zu verwenden ist und in bestehende Pipelines zu integrieren
 - Für jedes trainierte OVERFITSDF wird die gewählte Netzwerkarchitektur und die Geometrietransformationsmatrix werden als erste Bytes geschrieben

Methodik

- Die festen Speicherprofile und das Speicherlayout der unserer erlernten impliziten Funktionen bieten konsistente Abfrage- und Rendering Geschwindigkeiten.

Ergebnisse

- Performance
- Konvertierung
- Komprimierung
- Thingi10k



Performance

- 34 FPS bei 512x512 (Nvidia P100 GPU)
- “[...] not acceptable for real-time rendering applications.”
- Kann wie andere Implicits manipuliert werden

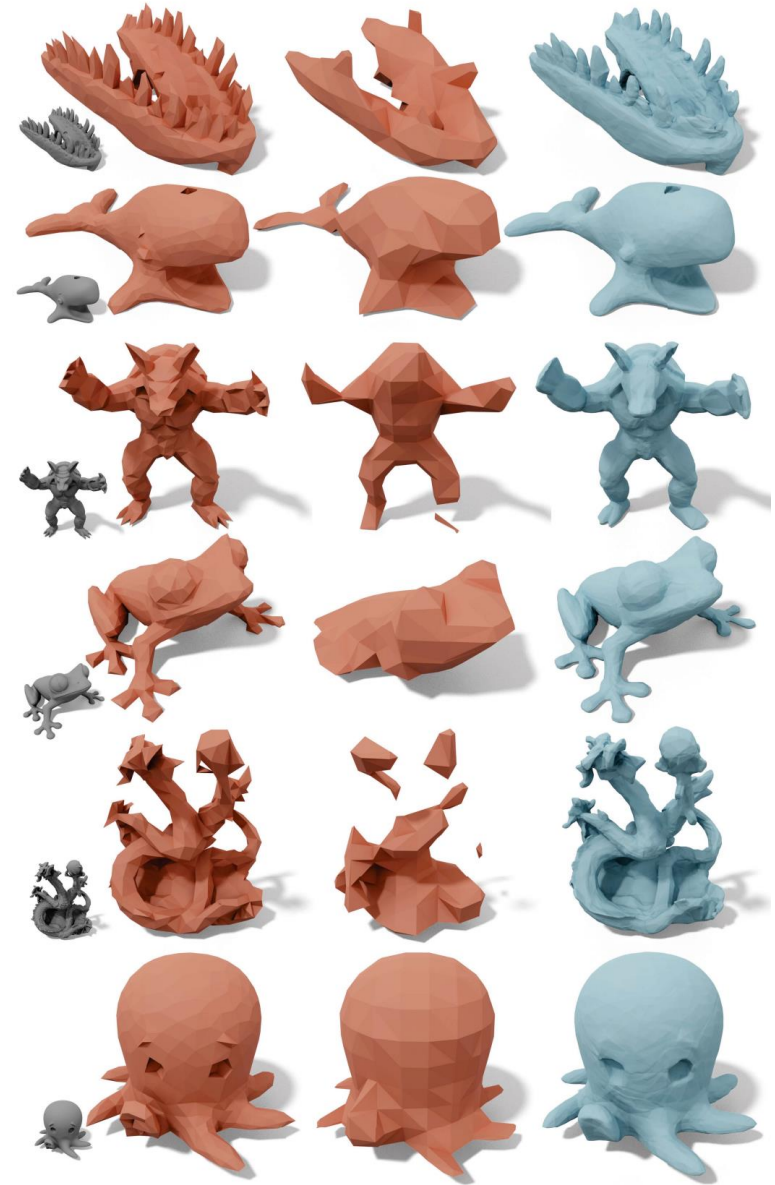


Konvertierung

- Bei minimaler Komplexität (8 layers of 32 neurons)
 - Durchschnittlich 90s für beliebige Geometrie
 - 64kB Speicher benötigt
- Thingi10k Dataset
 - Konvertiert in 16h (Nvidia Titan RTX) bzw 4h (4 Nvidia Titan RTX parallel)
 - Speichernutzung reduziert von 38.85 GB auf 640 MB
- Möglichkeit Komplexität basierend auf Input Geometrie zu ändern

Komprimierung

- bessere Annäherung
bei gleicher Speichernutzung
- 93% von Thingi10k
mit surface error unter 0.003
- Vergleich:
adaptive decimation (left)
uniform SDF (middle)
neural implicit (right)



Limitierung & Anwendungen

- Probleme bei minimaler Komplexität
- Noch nicht nutzbar für Echtzeit Anwendungen
- Reduzierte Speichernutzung
- Generelle Vorteile von Implicit

Limitierung

- Minimale Konfiguration nicht ausreichend bei komplexer Geometrie
 - Surface error beim Training prüfen
 - und nach Bedarf Komplexität anpassen
- Probleme bei Echtzeit Anwendungen
 - Vergleichsweise hohe Renderzeit
 - Zukunftspotential (hard- & software)

Anwendungen

- Reduzierte Speichernutzung
 - Langzeitspeicherung und Aufbewahrung
 - Große Projekte ohne Echtzeit Anforderung
- Vorteile von Implicit Representations
 - Praktisch unendliche Auflösung
 - Einfach zu manipulieren
 - Uniform memory pattern