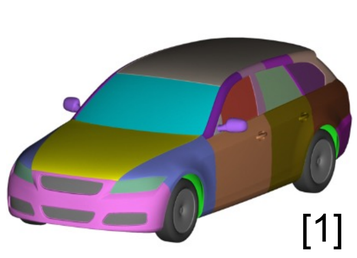
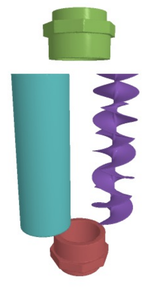
Learning Structured 3D Shape Representations for Design and Optimization

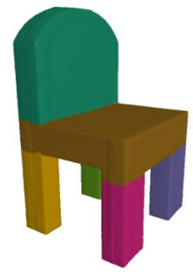
Nicolas Talabot
EPFL, CVLab

Why Does This Matter?

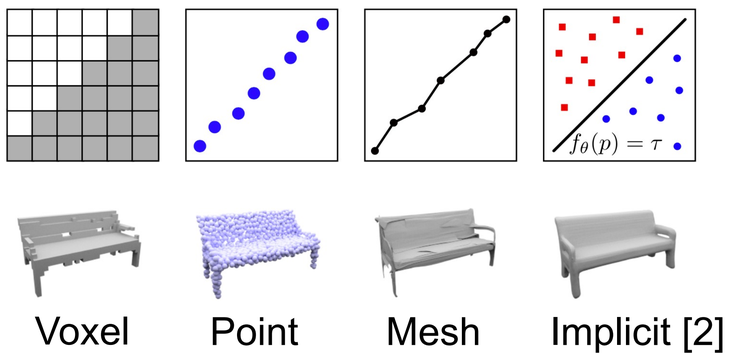
- Many objects are **composed of multiple parts** that must satisfy **physical and design constraints**.
- Modern Computer-Aided Design and Engineering require **differentiable, structured, and optimizable 3D representations** for automated design.


[1]



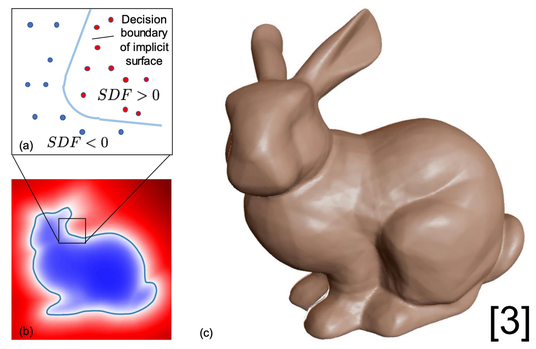


Implicit Shape Representation



- Define a shape as a **continuous function** rather than an explicit surface.
- Allow for **infinite resolution, continuous, and smooth topology changes.**
- We use **signed distances** for a smooth representation.
- Can be meshed to retrieve a surface. [4]

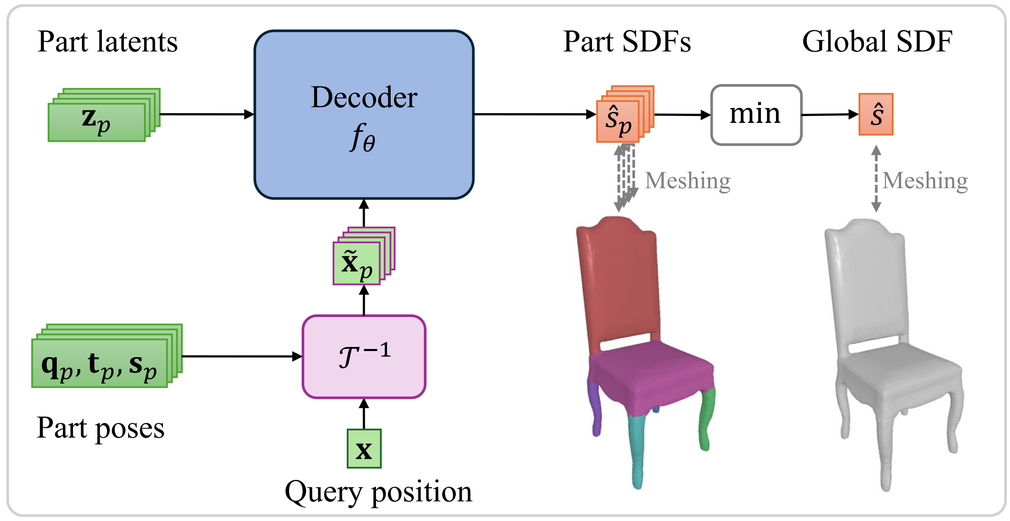
Signed Distance Function



[3]

PartSDF: Part-Based Implicit Neural Representation

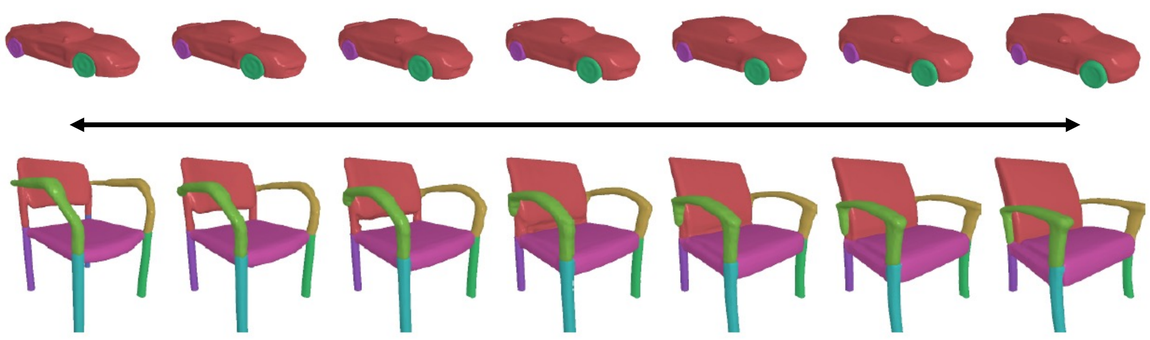
- **Model objects as distinct parts**, each represented as an **implicit SDF**.
- A **shared decoder** reconstructs the full shape from **per-part latent vectors and poses**.
- Ensures **smooth integration of parts** while maintaining **global consistency**.
- Supports **part-based optimization**, enabling integration with **CAE models** for real-world engineering applications.



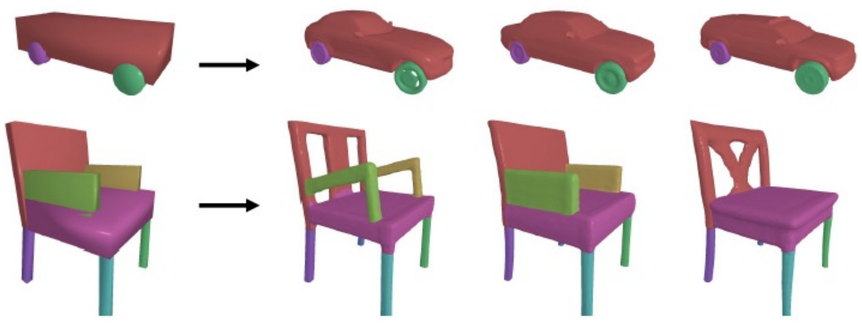
What Can We Do With it?

- The simplicity of the method makes its **modular**, with various uses.
- In all cases, the **global structure** of the shape and the **consistency between parts** are preserved.

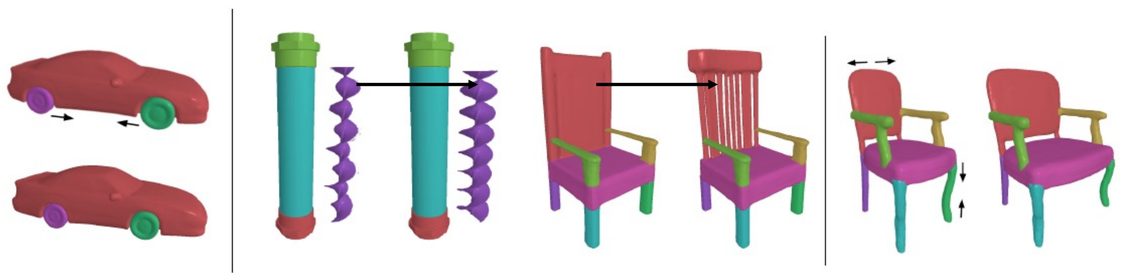
Interpolation → Smoothly blend two shapes



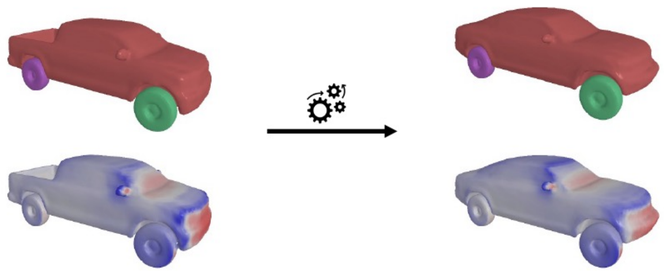
Generation → Synthesize diverse new objects



Manipulation → Modify individual parts



Optimization → Refine design for aerodynamism



References

[1] DrivAerNet++, Elrefaie et al., 2024
[2] Occupancy Networks, Mescheder et al., 2019
[3] DeepSDF, Park et al. 2019
[4] Marching Cubes, Lorensen and Cline, 1987

PartSDF: github.com/cvlab-epfl/PartSDF