Spatial Intelligence

The Next Frontier for AI

7 min read · Just now

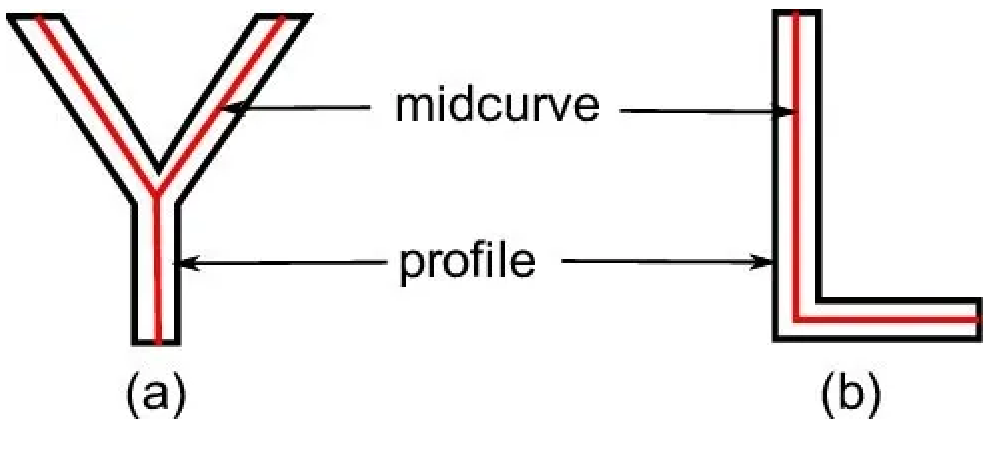
Yogesh Haribhau Kulkarni (PhD)

Photo by Susan Q Yin on Unsplash

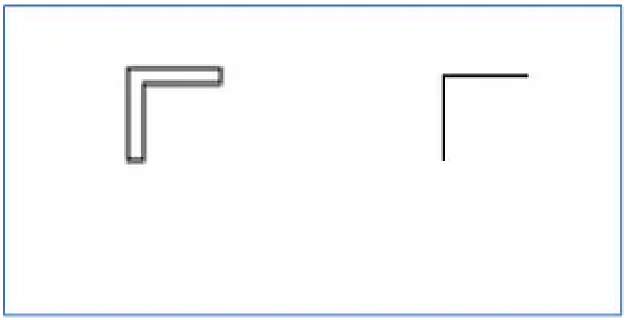
AI's recent breakthroughs in language and vision have been transformative, but they operate on 1D sequences or 2D pixel grids. Real-world intelligence demands true spatial understanding, an inherently geometric, physics-rich domain. As Fei-Fei Li notes, "Language literally comes out of everybody's head… But the world is far more complex." Unlike linear text, physical space is *"projected, lossy, and… mathematically ill-posed."* Cameras collapse 3D into 2D, forcing AI to *"reconstruct an inherently incomplete view,"* a task that humans solve via depth perception and physics priors. Li argues that spatial reasoning isn't just harder than language, it's **more important** for real intelligence. In short, **spatial intelligence is now seen as essential for any path to AGI/ASI.** AI systems that can reason about location, shape, and geometry could transform fields from robotics to virtual world design, but today's models still struggle.

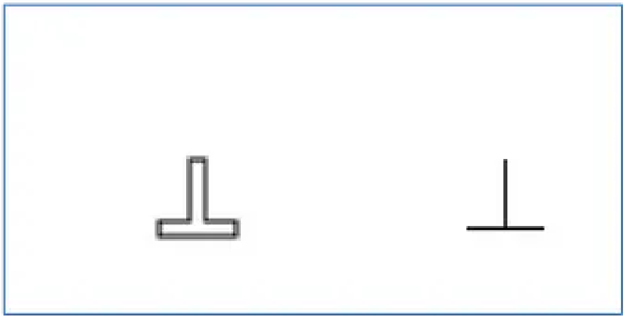
Challenges of Geometric Data in AI

Geometric data i.e. points, lines, curves, meshes, resists the "flattening" that models typically rely on. Consider a 2D shape defined by vertices: it isn't naturally a sequence like a sentence. For example, you can't trace a shape like a donut in one continuous line without lifting your pencil. Thus standard sequence-to-sequence (Seq2Seq) neural nets fail. Even graph neural networks (GNNs), which excel at relational data, hit limits: classic graphs capture **topology** (connectivity) but not **geometry** (exact coordinates or curved arcs). In a shape, each node has coordinates and edges have geometric length or curvature. A GNN would need to *"convolute at nodes (with coordinates) around arcs (with geometry)",* a capability still in research. Pooling such information into fixed-size vectors without losing spatial detail is an open problem.



Another hurdle is **variable length**. A shape's outline might have *m* points, but its skeletal mid-curve has *n* points, and m and n generally differ. Moreover, the output mid-curve may have branches or loops that the input loop does not. Standard neural encoders/decoders assume fixed-size inputs, so one cannot simply pad coordinates (a “(0,0)” pad might itself be valid!). In effect, this is a *“network-to-network”* translation problem, not a simple sequence mapping. AI researchers typically dodge this by converting geometry to images. For instance, a 2D shape can be rendered as a black-&-white bitmap and fed into a convolutional encoder; the mid-curve target is another bitmap. Image-based methods allow fixed-size tensors and use architectures like Pix2Pix or U-Net. But this **“dilution” to images** introduces approximation. Rasterization loses exact coordinates, and the neural output (a fuzzy pixel map) must be post-processed back to precise geometry.





In summary, geometric graphs defy standard encoders/decoders because they combine variable topology, variable scale, and precise continuous coordinates. As one survey notes, applying deep learning to **CAD’s parametric geometric data is still rare**, because converting solid geometry (B-Rep) to meshes either blows up memory or loses detail. Hence, breakthroughs require new representations and models.

MidcurveNN: A Novel 2D→1D Transformation

An exciting direction tackling these issues is the **MidcurveNN** project. Its goal is simple to state but complex to solve: “Given a 2D closed shape (a polygon), find its mid-curve (a 1D polyline skeleton)”. This is essentially *graph summarization or dimension reduction*: compress a 2D shape graph to a 1D graph that preserves its form. Traditional algorithms (medial axis transform, thinning, straight skeletons) exist, but they often produce noisy branches or require manual fixes. MidcurveNN seeks a learning-based solution.

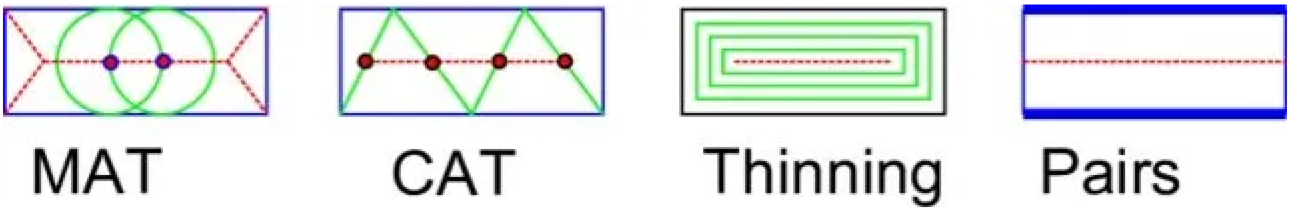
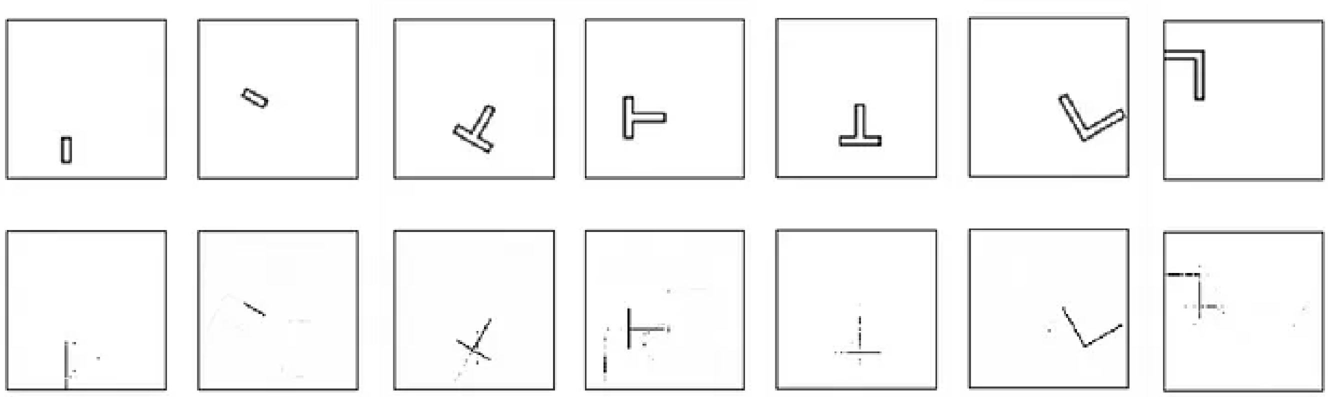


Image-to-Image Learning: Here both profile and mid-curve are treated as images. The 2D shape is rasterized to a 100×100 bitmap, and the target mid-curve drawn in another bitmap. An encoder-decoder (e.g. U-Net/Pix2Pix) is trained on pairs. This sidesteps geometry issues: images are fixed-size and amenable to CNNs. Data is augmented by shifting, rotating, and scaling the shapes. Early results show that a simple one-layer encoder-decoder *"can learn the dimension reduction function reasonably well"* on simple shapes. However, outputs can be noisy at boundaries and work best on shapes similar to the training set.



Text-to-Text Learning: A key innovation is to serialize geometry in text form (JSON) similar to CAD's boundary representation (B-Rep). For example:

```
{
  "ShapeName":"I",
  "Profile":[(5,5),(10,5),(10,20),(5,20)],
  "Midcurve":[(7.5,5),(7.5,20)],
  "Profile_brep":{ "Points":[(5,5),(10,5),(10,20),(5,20)],
                   "Lines":[[0,1],[1,2],[2,3],[3,0]],
                   "Segments":[[0,1,2,3]] },
  "Midcurve_brep":{ "Points":[(7.5,5),(7.5,20)],
                    "Lines":[[0,1]],
                    "Segments":[[0]] }
}
```

This text-based B-Rep lays out exact points and edges for both shape and mid-curve. By representing shapes in machine-readable CAD-like form, one can leverage powerful language models. Given a dataset of shapes serialized in the above JSON/B-Rep format, one can fine-tune LLMs (e.g. via HuggingFace) to *"translate"* from profile to mid-curve descriptions. Early experiments (see "Geometry, Graphs and GPT") explore whether GPT-style models can learn to predict the mid-curve from a sequence of coordinates. Text methods have the advantage of **precision**: unlike pixel outputs, the LLM's answer would list exact coordinates (any erroneous lines could be pruned algorithmically). The challenge is crafting a good "language" of graphs; current graph-to-text encodings (surveyed in *"Talk like a graph"*) aren't geometry-specific, so the B-Rep format is a proposed custom solution.

Graph-to-Graph Learning: The next step is to operate directly on geometric graphs. Both input and output are represented as polyline graphs: nodes have (x,y) coordinates, edges link them. The idea is to build a U-Net–like network where standard 2D convolutions are replaced by *graph convolutional layers* (e.g. from Deep Graph Library). This "geometry-to-geometry" encoder-decoder would learn from many input-output polygon pairs generated by geometric transformations. Preliminary work is exploring variational graph auto-encoders to compress this graph information.

Why MidcurveNN Is Promising?

MidcurveNN represents a **novel integration** of geometry and AI. It tackles the classic CAD/CAE problem of midsurface/midcurve generation through modern deep learning. By combining image-based learning with direct graph convolution and even LLMs, it pioneers new ground. As Boussuge et al. note, the closest previous work uses CNNs on images to clean up skeletons, but these require high-resolution grids and struggle on complex shapes. In contrast, MidcurveNN's multi-modal strategy can potentially generalize better and produce clean outputs. Early results (on simple "L" shapes) already recover reasonable mid-curves, and ongoing work aims to handle arbitrary shapes.

Impact on CAD, CAE, and Beyond

Computer-Aided Design & Engineering: Dimension reduction is ubiquitous in CAD/CAE. Thin-walled solid parts (like car body panels) are often collapsed to a *midsurface* for fast finite-element simulation. In 2D, a planar profile is reduced to a *mid-curve* (or *skeleton*) for similar reasons. Traditionally this uses geometric constructs (medial axes, Voronoi diagrams, thinning). But these can be brittle; even commercial CAE tools often require manual edge pairing to fix broken skeletons. A learning-based MidcurveNN could automate skeletonization and midsurface extraction robustly. This promises faster design iterations, automated rule extraction, and integration with generative design tools.

Broader Graph Transformations: The problem of mapping one graph to another appears in many networks beyond geometry. Social, biological, or infrastructure networks sometimes need *summarization* or *relabeling* while preserving structure. MidcurveNN's approach, particularly its text/LLM angle, points to a future where large language or graph-transformer models can handle general graph translation tasks. The notion of encoding a graph's geometry with **Spatial Attention** may inform tools for program synthesis, network compression, or knowledge graph reasoning.

Relevance to AI/ML Research: This work highlights a key gap in AI capabilities. As **Geometric Deep Learning** surveys note, applying neural nets to CAD's exact shapes is largely uncharted. Boundary Representations (B-Reps) pack rich geometry but don't easily fit into CNN pipelines. MidcurveNN shows one path forward: blending raster-based deep learning with vector-text representations and GNNs. In doing so, it is at the intersection of neural networks, symbolic/structured data, and traditional CAD.

Conclusion

In summary, spatial and geometric reasoning remain a frontier challenge for AI. Problems like extracting mid-curves from shapes exemplify why: they demand variable-length, continuous, and spatially-rich outputs from inputs that are topological. Standard neural models falter on such tasks. MidcurveNN is an emerging research direction that addresses these issues head-on, using multi-modal transformations (images, graphs, text) and novel representations (B-Reps). Its early success underscores the promise of combining classic geometric insight with modern deep learning. As AI seeks to reach AGI/ASI levels, developing true spatial understanding through projects like MidcurveNN will be crucial for fields from industrial design to robotics.

Spatial Intelligence | Artificial Intelligence | Geometry | Neural Networks | AGI

Following

Published in Technology Hits

3.6K followers · Last published just now

We cover important, high-impact, informative, and engaging stories on all aspects of technology. Subscribe to our content marketing strategy newsletter: https://drmehmetyildiz.substack.com/ Writer inquiries: https://digitialmehmet.com/contact

Edit profile

Written by Yogesh Haribhau Kulkarni (PhD)

1.8K followers · 2.1K following

PhD in Geometric Modeling | Google Developer Expert (Machine Learning) | Top Writer 3x (Medium) | More at https://www.linkedin.com/in/yogeshkulkarni/

No responses yet

Yogesh Haribhau Kulkarni (PhD)

What are your thoughts?

More from Yogesh Haribhau Kulkarni (PhD) and Technology Hits