Discontinuity-Aware 2D Neural Fields

Yash Belhe - University of California San Diego
Michael Gharbi, Matthew Fisher, Iliyan Georgiev - Adobe
Ravi Ramamoorthi, Tzu-Mao Li - University of California San Diego
Path-tracing can produce arbitrarily high resolution images
Let’s re-render a 100x close-up here
100x zoom — image has discontinuities!
Discontinuity locations are analytically known
Most image formats do not use discontinuity information
Our contribution

Hybrid neural-mesh-based representation for images

• Is optimizable
• Can be rendered at any zoom scale in real time
• Can preserve discontinuities that are given
Common image representations
Raster images can represent complex signals
... but details are limited by resolution
Neural fields can compactly encode giga images!

InstantNGP: Muller 22
... but they blur discontinuities

InstantNGP: Muller 22
Neural fields are continuous by construction
Neural fields are continuous by construction
Neural fields are continuous by construction

Feature field

Bilinear interpolation
Neural fields are continuous by construction

Feature field → Bilinear interpolation → Interpolated Feature
Neural fields are continuous by construction
Neural fields are continuous by construction

Feature field

Bilinear interpolation

Interpolated Feature

MLP

Color
Neural fields are continuous by construction

Feature field → Bilinear interpolation → Interpolated Feature → MLP → Color
Vector graphics analytically store discontinuity locations
... but they have simplistic shading
Our goal
Our goal: encode a target image
Our goal: encode a target image given its discontinuity locations
Curved discontinuities
Feature field construction
Feature field construction

Discontinuity locations
Feature field construction

Discontinuity locations → Triangulation

Hu 19: TriWild: Robust Triangulation with Curve Constraints
Feature field construction

Discontinuity locations → Triangulation → Feature field

Hu 19: TriWild: Robust Triangulation with Curve Constraints
Our feature field is aligned with discontinuities
Our rendering pipeline
Mapping queries to colors
Mapping queries to colors

Feature field
Mapping queries to colors

Feature field

Discontinuity-aware feature interpolation
Mapping queries to colors

Feature field

Discontinuity-aware feature interpolation

Interpolated Feature
Mapping queries to colors

Feature field

Discontinuity-aware feature interpolation

Interpolated Feature

MLP
Mapping queries to colors

Feature field

Discontinuity-aware feature interpolation

Interpolated Feature

MLP

Color
Discontinuity-aware feature interpolation
Continuous vertex
Continuous vertex
Continuous vertex
Continuous vertex
Discontinuous vertex
Different features above and below each discontinuity
Evaluating vertex feature for a query point
Closest clockwise feature
Closest counter-clockwise feature
Vertex feature = radially interpolate closest features
Closest features change on other side of discontinuity
Closest features change on other side of discontinuity
Closest features change on other side of discontinuity
Closest features change on other side of discontinuity
Putting it all together
Query point
Find triangle that contains query point
Zooming in to query point
Directly retrieve feature for continuous vertex
Retrieve features for discontinuous vertices
Find closest features
Radially interpolate closest features
Retrieve features for discontinuous vertices
Find closest features
Radially interpolate nearest features
Barycentrically interpolate three vertex features
Barycentrically interpolate three vertex features
Decode interpolated features using MLP

Interpolated feature
Decode interpolated features using MLP

Interpolated feature → MLP
Decode interpolated features using MLP
Recap

Feature field

Interpolated Feature

MLP

Color

Discontinuity-aware feature interpolation
Performance
60 FPS inference @1080p

- 60-120 FPS inference on our examples
- Training is typically < 2 mins

All numbers are reported on an RTX 3090Ti
Results
Application: path-traced images
Application: path-traced image
Application: path-traced image

Ours (1× zoom)

Ours (33×)

Ours (100×)

ReLU fields (100×)

InstantNGP (100×)
Application: diffusion curve images
We start with some curves
Colors on both sides of curves
Diffuse colors from curves
Diffusion curve image
Our result: curved discontinuities
Our result: open edges
Application: physics-informed diffusion curve
Application: physics informed diffusion curves
Application: physics informed diffusion curves

Walk-on-spheres reference

Ours

PSNR: 26.90 dB
Application: physics informed diffusion curves

- **Walk-on-spheres reference**
- **TanH fields**
  - PSNR: 17.49 dB
- **Ours**
  - PSNR: 26.90 dB
- **Multi-layer perceptron**
  - PSNR: 12.09 dB
- **SIREN**
  - PSNR: 18.57 dB
Application: store FEM solutions
Application: store solution to Helmholtz equation
Application: store solution to wave equation
Limitations

- We require discontinuity locations
- Different data structure needed for high frequency continuous variation
Converting an image to pixels requires choosing a resolution and throwing away information beyond that resolution... When you really think about it, representing an image as pixels is really a bad compression technique... we need better image atoms...

Jim Blinn’s Corner Notation Notation Notation Notation Notation

yashbelhe.github.io

Code available!