Occupancy Fields

An **occupancy field** is a function that represents whether a point in 3D space is inside, outside, or on the surface of an object where a higher value is inside the object.

$$F(x, y, z) : \mathbb{R}^3 \to [0, 1]$$

Fourier Occupancy Field

- Conference: NeurIPS 2022
- Authors: Qiao Feng, Yebin Liu, Yu-Kun Lai, Jingyu Yang, Kun Li
- Paper: https://arxiv.org/pdf/2206.02194
- GitHub: https://github.com/fengq1a0/FOF

Instead of storing full 3D occupancy values, the 3D object is represented as a 2D field and uses Fourier series expansion for the depth. The occupancy field $F(x, y, z)$ is converted into $H \times W$ 1D occupancy signals $f(z)$ along lines orthogonal to the 2D field.

$$f(z) = \frac{a_0}{2} + \sum_{n=1}^{N} (a_n \cos(n\pi z) + b_n \sin(n\pi z))$$

where a and b are the learnable coefficients that represent the 3D occupancy at a 2D location.

The final Fourier Occupancy Field (FOF) is stored as a multi-channel 2D map with shape $H \times W \times (2N + 1)$. Since FOF uses a 2D CNN to estimate the field, it loses fine details and edge information. Also, they mention that FOF cannot represent objects that are too thin, such as fingers.

Diffusion-FOF

- Conference: CVPR 2024
- Authors: Yuanzhen Li, Fei Luo, Chunxia Xiao
- Paper: Paper

Method

1. A **Siamese network training strategy** is used to train a deep neural network to estimate the back-view image from the front-view image and ensure style consistency between the views.
2. An initial **Fourier Occupancy Field** (FOF) is predicted from the image.
3. The method first applies a **Haar wavelet transform**, decomposing the FOF into four wavelet bands, representing the average of the source image, which captures the smooth global structure, and fine details in the vertical, horizontal, and diagonal directions.

$$\{A, V_1, V_2, V_3\} \in \mathbb{R}^{\frac{H}{2} \times \frac{W}{2} \times (2N+1)}$$

4. Since the initial FOF lacks fine details and thin objects, a **diffusion** model is trained to refine the FOF and enhance fine details such as fingers.
5. The Inverse Haar wavelet transform reconstructs the full FOF with the fine details from diffusion.
6. The FOF is transformed into occupancy values and the **Marching Cubes** algorithm generates the final polygonal 3D mesh from the values.

Future Work

Instead of querying a distance from the SDF decoder, the hand and object pose features can be directly extracted from the Diffusion-FOF. SDF requires querying many points to build the surface, while FOF stores the whole field that can be directly used. Also, SDF only represents the closest surface to a query point, but Diffusion-FOF models the full volumetric occupancy field.

Neural Radiance Fields

A neural radiance field takes in 5D coordinates, including the spatial location (x, y, z) and viewing direction (θ, ϕ), and outputs the view-dependent color and volume density at that point and view. The volume density represents how much light is accumulated at that point, where high density is more likely to be part of a solid object and a low density means the point is in free space. The volume density can also be thought of as the probability of a ray terminating at that point.

- Conference: ECCV 2020
- Authors: Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, Ren Ng
- Paper: https://arxiv.org/pdf/2003.08934
- GitHub: https://github.com/bmild/nerf

Method

1. The camera intrinsic parameters are used to convert each 2D pixel coordinate from the image into the 3D coordinate in the world space.
2. A ray is created from the camera center to each of the 3D coordinates using the camera extrinsic parameters, and then a set of 3D coordinates are sampled along each camera ray.
3. Each 3D coordinate is passed through an 8-layer MLP to obtain the volume density and a feature vector. The feature vector is concatenated with the corresponding 2D viewing direction from the ray and passed through another fully connected layer to determine the color.
4. The model performs an initial pass with uniform sampling, but many of those points may fall in empty space which don't contribute to the final pixel value. The density values are used to build a probability distribution. An additional set of points are sampled from this distribution, where more points are sampled from regions with high density.
5. Before passing the coordinate into the MLP, a positional encoding is applied to map the coordinate into a higher dimensional space.

   $$\gamma(p) = (\sin(2^0\pi p), \cos(2^0\pi p), \ldots, \sin(2^{L-1}\pi p), \cos(2^{L-1}\pi p))$$

   where L is a hyperparameter.
6. For each ray, the final pixel color from the camera's perspective is computed as a weighted sum using the densities and colors from the sampled points, considering occlusions, transparency, and view-dependent effects.

Hamba

- Conference: NeurIPS 2024
- Authors: Haoye Dong, Aviral Chharia, Wenbo Gou, Francisco Vicente Carrasco, Fernando De la Torre
- Paper: https://arxiv.org/pdf/2407.09646

Method

1. The image is fed into a Vision Transformer backbone to extract a set amount of tokens, which are then downsampled using convolution layers.
2. The Joint Regressor, a stack of 2D Selective Scan (SS2D) blocks and an MLP head, regresses 3D joints and projects them back to 2D with a predicted camera translation, and aligns tokens with bilinear interpolation.
3. A Graph-guided Bidirectional Scan (GBS) refines the joint tokens to model both local and global joint dependencies using a Graph Convolutional Network (GCN).
4. The GBS-refined tokens are passed through the Graph-guided State Space (GSS) block, which applies a state-space model to propogate joint features across spatial and temporal relationships to enforce joint relationships.
5. The GSS-refined tokens, global mean feature, which aggregates information across all tokens, and joint predictions are fused and passed to an MLP to regress the final MANO parameters.

Future Work

A graph is defined where nodes represent hand joints and object keypoints and edges connect spatially relevant points like bones and object surface relationships. Each node would contain the feature vector, consisting of positional encodings and image features, and initial depth estimates. The depth for visible keypoints are obtained using a depth estimation network, while occluded depths are predicted by passing the visible keypoints through a Graph Convolution. A Graph-guided Bidirectional Scan (GBS) refines the node features to model both local and global joint dependencies using the Graph Convolutional Network (GCN). Graph Attention applied to learn relationships within the hand (hand-to-hand edges) and across the hand-object interaction space (hand-to-object edges). The same would be applied for object nodes to generate object-internal and cross-field object features.

1. **Graph Definition**

   Each node represents either a hand joint or an object keypoint. There are hand-to-hand edges that are predefined and follow the hand keypoint structure, object-to-object edges that are predifined and connect nearby object nodes, and hand-to-object edges that are learned and connect spatially neighboring hand and object nodes.

2. **Initial 3D Coordinate Prediction**

   The 2D keypoints from an off-the-shelf network are combined with the depth of visible keypoints from a depth estimation network. Visible keypoints are determined based on a threshold of the 2D keypoint confidence. A Unified GCN is used to propagate depth features from visible to occluded keypoints by aggregating spatially neighboring node information.

3. **Unified Graph-guided Bidirectional Scan**

   A backward scan refines the node features by incorporating global information, ensuring that occluded keypoints align with spatial relationships. The hand-object edges are dynamically adjusted through Graph Attention to improve structural consistency.