Neural Feature Fusion Fields

3D Distillation of Self-Supervised 2D Image Representations

Vadim Tschernezki, Iro Laina, Diane Larlus, Andrea Vedaldi
Outline

- Motivation
- Background
  - 2D Feature Extraction
  - Neural Rendering
  - Semantic NeRF
- Neural Feature Fusion Fields (N3F)
- Results
  - Qualitative & quantitative
- Limitations & future work
Motivation

- Each idea helps improving the other one
- Same input domain
→ combine both methods
Feature Extraction - DINO

- Operates in 2D
- Architecture based on Vision Transformer (ViT)
- Self-supervision
- Self-distillation
• Other 2D feature extractors:
  – MoCo-v3, DeiT
Neural Rendering - NeRF

1) 5D Input
Position + Direction

2) Output
Color + Density

3) Volume Rendering

4) Rendering Loss

\[ \| \text{g.t.} \|^2 \]

\[ \| \text{g.t.} \|^2 \]

[13]
NeRF - results

[15]
Semantic NeRF

**Input**
- RGB images
- Semantic labels

**Output**
- Semantic 3D reconstruction

[9]
Neural Feature Fusion Fields (N3F) - idea

Image sequence

Objects segmented via 3D-fused features

Self-supervised 2D features

2D to 3D

Object retrieval

3D to 2D

Scene editing

[3]
N3F - method

Teacher: Feature Extractor

DINO

Loss

\[
\Phi(I_{tu}) - \Phi(I_{tu})^2 + \Phi(I_{tu})^2 - \Phi(I_{tu})_u + \Phi(I_{tu})_u^2
\]

\[\hat{\Phi}_{tu} \]

N3F

Volume Rendering

Student: Neural Renderer

NeRF

2D Input: Image

3D Input: Scene
Two main contributions:

- Regularization of feature prediction
- Open-world knowledge

N3F - method

Final loss function:

$$\sum_t \left( \| \hat{I}_t - I_t \|^2 + \lambda \| \hat{\Phi} - \Phi(I_t) \|^2 \right)$$
Experiments
Experiments

Setup:

1) Simple static scenes (LLFF dataset):
   - N3F with NeRF

2) Egocentric videos (EPIC-KITCHENS):
   - N3F with NeuralDiff
NeuralDiff

Neural renderer for egocentric videos

NeRF
Applications

- Given: collection of images + query image patch

![Image: $I_t$](image1.png)

![Patch: $R_t$](image2.png)

Feature descriptor:

$$\Phi(I_t)_{R_t}^{avg} = \frac{1}{|R_t|} \sum_{u \in R_t} \Phi(I_t)_u$$

- 2D object retrieval
- 3D object segmentation
- Scene editing
- Amodal segmentation

[3]
## 2D object retrieval - quantitative results

<table>
<thead>
<tr>
<th>Method</th>
<th>S01</th>
<th>S10</th>
<th>Average (abs gain)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DINO [4] [ViT-B/8]</td>
<td>75.75</td>
<td>65.79</td>
<td></td>
</tr>
<tr>
<td>N3F (DINO)</td>
<td>83.64</td>
<td>82.17</td>
<td><strong>76.26</strong> (+11.91)</td>
</tr>
<tr>
<td>DINO [4] [ViT-B/16]</td>
<td>77.37</td>
<td>62.19</td>
<td>60.15</td>
</tr>
<tr>
<td>N3F (DINO)</td>
<td>88.61</td>
<td>84.93</td>
<td><strong>78.84</strong> (+18.69)</td>
</tr>
<tr>
<td>MoCo-v3 [8] [ViT-B/16]</td>
<td>70.73</td>
<td>60.12</td>
<td>57.51</td>
</tr>
<tr>
<td>N3F (MoCo-v3)</td>
<td>86.67</td>
<td>83.21</td>
<td><strong>76.15</strong> (+18.64)</td>
</tr>
<tr>
<td>DeiT [58] [ViT-B/16]</td>
<td>55.27</td>
<td>52.48</td>
<td>47.51</td>
</tr>
<tr>
<td>N3F (DeiT)</td>
<td>86.02</td>
<td>83.12</td>
<td><strong>74.53</strong> (+26.82)</td>
</tr>
</tbody>
</table>

[3]
2D object retrieval – qualitative results

- Calculate feature values of target images
- Match with query feature descriptor
3D object segmentation

- N3F features can be used “out of the box”
- Segment every point that matches queried mean feature descriptor
Scene editing

- Remove items from images
- Same as 3D object retrieval
  - additionally: set volume density to zero

![Images showing scene editing examples]
Amodal segmentation

- Segmenting full extent of an object
- Similar as 3D object retrieval
- Set occupancies to zero for regions that are dissimilar
Limitations & Future work
Limitations

• Upper bound of performance
• Self-supervised features group related objects
• Reliability on unsupervised machine learning
• Geometry errors by neural renderer
Future work

• Cross-instance correspondence
• Applying N3F in different setups
  – Feature extractor & neural renderer
  – Application domain
• Overcoming time & memory constraints
References


References


