Clonoxels: Exploiting Repetitive Structures for Efficient View Synthesis

Oliver Camilleri
https://www.surrey.ac.uk/people/ollie-camilleri

Centre for Vision, Speech and Signal Processing (CVSSP),
University of Surrey

Machine Learning is rapidly becoming a staple tool in media production pipelines. In particular, learned 3D representations have seen an explosion of interest through methods like neural radiance fields (NeRFs) [3]. These techniques enable photorealistic novel view synthesis (NVS), convincingly rendering scenes from new camera angles. Such methods build representations directly from images, reconstructing real-world environments without reliance on handcrafted assets. This opens new possibilities in set digitization and immersive media.

NeRFs and related *implicit* methods have now been surpassed in popularity by *explicit* techniques that include voxel-based approaches such as Plenoxels [5] and point-based representations like 3D Gaussian Splatting (3DGS) [2]. Although they reduce training and inference times considerably, these techniques are associated with large memory footprints and their NVS capacity relies on extensive multi-view capture. Sparse coverage or occlusion leads to under-constrained optimization, often producing noisy artifacts and degraded visual quality. In this paper, we attempt to address these issues by pooling information from repeated objects which are typically visible from a range of viewpoints. This better constrains reconstruction and allows regions to share information, saving memory. Concurrent research has used this principle to enhance the NVS of 3DGS [4] yet relies on manual object selection and does not explicitly explore memory reduction.

In contrast, we propose Cloning-Plenoxels, or Clonoxels, a voxel-based framework that automatically detects and shares information between repeat objects. We directly measure storage savings and use an alternative, more memory-conscious approach to handling lighting differences. Voxels avoid the overhead of a large neural network while providing a truly volumetric representation whose regularity makes defining discrete objects practical. We build directly on Plenoxels, discretizing scene space into a grid structure and storing learned features at vertices. Each vertex encodes scene properties which are queried via interpolation and converted into images using emission-absorption volume rendering. Repeated objects are identified in training images using Mask2Former [1] as it offers high-quality instance and semantic segmentation. For efficiency, detected semantic classes are ranked by instance count - we only keep masks corresponding to the most frequent class. These are then projected into the scene to localize objects in 3D, with voxels assigned by multi-view consistency. We refer to each such isolated region as an object subgrid. Given an object's feature subgrid, $\mathcal{F}$, we apply a learnable rigid transformation to each vertex, $\mathbf{v}$. The transformation is parameterized by a rotation matrix $\mathbf{R} \in \mathrm{SO}(3)$, derived from a learnable quaternion, a center of rotation $\mathbf{o} \in R^3$, and a translation vector $\mathbf{t} \in R^3$:

$$\mathbf{v}' = \mathbf{R}(\mathbf{v} - \mathbf{o}) + \mathbf{o} + \mathbf{t} \tag{1}$$

Using PyTorch's differentiable grid_sample, the transformed grid coordinates are used to resample features, producing a transformed subgrid, $\mathcal{F}'$. The transformation parameters $(\mathbf{R}, \mathbf{o}, \mathbf{t})$ are optimized via backpropagation through a loss that compares $\mathcal{F}'$ to the subgrid associated with another instance of the same object, $\hat{\mathcal{F}}$:

$$\mathcal{L}_{feat} = \lambda_{feat} ||\mathcal{F}' - \hat{\mathcal{F}}||_1. \tag{2}$$

This comparison is weighted by a coefficient, λ_{feat}. L1 is used over L2 due to its outlier tolerance, increasing robustness to noisy reconstructions. Sharing information in this way does not tolerate differences in lighting. To allow for such variations, like shadows or bright, specular highlights, we introduce a view-dependent color correction in the form of a scalar voxel feature, Δc, which is not subject to $\mathcal{L}_{feat}$ and can therefore differ between objects. We do not allow for per-channel corrections and instead apply them equally to all three color channels. This limitation helps maintain the memory-saving advantage of our approach and encourages the sharing of as much appearance information as possible. The color associated with a given subgrid voxel is then defined by:

$$\mathbf{c}' = \mathbf{c} + \Delta c \cdot \mathbf{1} \tag{3}$$

Where $\mathbf{1} = (1,1,1)^{\mathrm{T}}$. Clamping ensures that $\mathbf{c}'$ remains in the range [0,1].

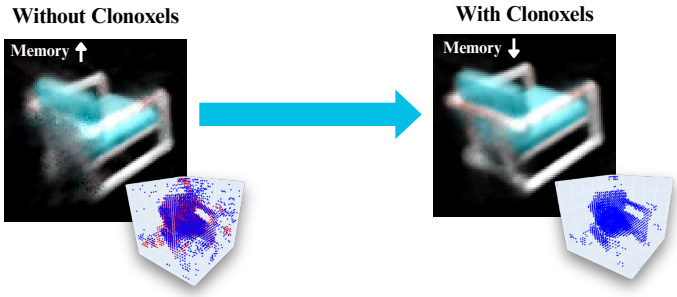

Figure 1: Clonoxels recovers structure and appearance when rendering from occluded views. The point clouds show overlaid object subgrids and demonstrate that the extra reconstruction constraints denoise 3D representations.

We optimized two real and two synthetic scenes using an NVIDIA GeForce RTX 3090. Fig. 1 shows the successful recovery of an object rendered from a view that was obstructed during training. Here, **drawing information from other objects reduced the mean squared error (MSE) by 41%**. Furthermore, in all cases, **information exchange resulted in denoised 3D geometry**. This is likely due to the additional reconstruction constraints that Clonoxels provides. Such denoising is particularly promising, as voxel-based representations lend themselves well to mesh model extraction. One synthetic scene, exhibiting sharp lighting, was reserved for testing the effect of Δc. Despite being broadcast to all three color channels, and avoiding any additional parameters or losses, it improved image reconstruction substantially, lowering MSE by 44%. Within the remaining scenes, we found that **Clonoxels saved an average of 44.3MB**.

While these results are encouraging, several avenues for improvement remain. Clonoxels relies on high-quality semantic and instance segmentation. This component is modular, however, and can be easily advanced via emerging segmentation models. Additionally, semantically similar objects may exhibit significant visual differences, compromising the gains from feature sharing. We consider this an interesting direction for future work that could include weighting the similarity loss using a perceptual metric that considers image features. It may also be possible to combine Clonoxels with generative in-filling approaches to more faithfully reconstruct scenes from unseen viewpoints.

Overall, we believe that this general principle of intra-scene information exchange remains under-studied. It appears to have great potential to deliver substantial 3D reconstruction enhancements by lowering storage demands and easing constraints on data acquisition via the recovery of information in the face of visual occlusion.

[1] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention Mask Transformer for Universal Image Segmentation. *arXiv e-prints*, art. arXiv:2112.01527, December 2021. doi: 10.48550/arXiv.2112.01527.

[2] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *arXiv e-prints*, art. arXiv:2308.04079, August 2023. doi: 10.48550/arXiv.2308.04079.

[3] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. *arXiv e-prints*, art. arXiv:2003.08934, March 2020. doi: 10.48550/arXiv.2003.08934.

[4] Nicolás Violante, Andreas Meuleman, Alban Gauthier, Frédo Durand, Thibault Groueix, and George Drettakis. Splat and Replace: 3D Reconstruction with Repetitive Elements. *arXiv e-prints*, art. arXiv:2506.06462, June 2025. doi: 10.48550/arXiv.2506.06462.

[5] Alex Yu, Sara Fridovich-Keil, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance Fields without Neural Networks. *arXiv e-prints*, art. arXiv:2112.05131, December 2021. doi: 10.48550/arXiv.2112.05131.