# Convolutional Nearest Neighbors: Reinterpreting Convolution Through K-Nearest Neighbor Selection

**Mingi Kang, Jeova Farias**

Bowdoin College
mkang2@bowdoin.edu, j.farias@bowdoin.edu

## Abstract

Convolutional operations have served as a foundational component of deep learning over the past three decades, driving significantly advances in computer vision through convolutional neural networks (CNNs). Despite their widespread adoption, the relationship between convolutions and k-nearest neighbor algorithms has not been explicitly explored in existing literature. This work introduces Convolutional Nearest Neighbors (ConvNN), a novel operation that reinterprets convolution through the k-nearest neighbor selection principle.

## Related Work

The k-nearest neighbors (k-NN) algorithm, though classical, remains influential in deep learning. It is often used to evaluate learned embeddings, where clustering of semantically similar samples indicaes representation quality (Wu et al., 2018). In self-supervised learning, k-NN enables non-parametric classification and retrieval without training extra classifiers (He et al., 2020; Chen et al., 2020). It also underpins models for point clouds and manifolds, where local neighborhoods drive convolution-like operations (Qi et al., 2017), and inspires hybrid approaches such as Neural Nearest Neighbors Network (N3Net), which integrate k-NN search into trainable architectures (Plötz & Roth, 2018). Beyond evaluation approximate k-NN powers large-scale retrieval and few-shot learning, making it a practical building block in modern pipelines.

## Methodology

Traditional convolution can be reinterpreted as a k-nearest neighbor operation where neighbor selection is based purely on spatial distance between input features. A 3x3 convolutional kernel selects the $k = 9$ nearest neighbors surrounding each feature (itself included as a neighbor) based on Euclidean distance in image coordinates. These spatially defined neighbors are then weighted and aggregated using learned parameters.

This spatial constraint, however, represents only one possible approach to neighbor selection. Feature-based similarity offers an alternative criterion for identifying nearest neighbors. Rather than restricting the aggregation to spatially adjacent features, we can identify and aggregate neighbors based on feature similarity, regardless of their spatial location within the input grid. This approach enables the aggregation of semantically similar regions that may be spatially distant from each other.

Algorithmically, our ConvNN algorithm generalizes neighbor selection in convolutions beyond fixed spatial patterns through three fundamental steps.

### Similarity Computation

Pairwise similarities are computed between all spatial positions in the feature map. Unlike standard convolution's implicit spatial similarity, we investigate multiple similarity metrics including Euclidean distance and cosine similarity in feature space. Given a feature map $X \in \mathbb{R}^{B\times C\times H\times W}$, spatial dimensions are flattened and similarities computed betwen all $H \times W$ positions.

### K-Nearest Neighbor Selection

Each position's k-nearest neighbors are identified through hard top-k selection based on computed similarities. This approach contrasts with standard convolution's fixed spatial neighbor selection, providing interpretable neighbor assignments based on feature values. The parameter $k$ directly controls the receptive field size, analogous to kernel size in standard convolutions.

### Weighted Aggregation

Features from selected neighbors are aggregated using learnable weights applied through 1D convolution with kernel size $k$ and stride $k$. This design maintains parameter efficiency while enabling content adaptive receptive fields that vary based on input features rather than remaining fixed to spatial locations.

### Computational Remarks

Three sampling strategies address the computational complexity of the $O(N^2)$ similarity computation and comparisons. Random sampling selects a random subset of positions on the input grid as potential neighbors. Spatial sampling creates a regular large resolution grid of reference points on the original input grid, maintaining spatial structure while reducing computational cost. Pixel shuffling (Shi

et al., 2016) reorganizes spatial dimensions into channel dimensions, effectively reducing spatial resolution before neighbor selection. This also contributes to the overall effectiveness of ConvNN as it considers patches of features instead of individual ones when computing neighbors.

## Results and Discussion

ConvNN demonstrates that convolution can be interpreted as k-nearest neighbor aggregation. When similarities are computed based solely on spatial distance, ConvNN reduces to standard convolution. When similarities are computed from feature values, ConvNN becomes a weighted aggregation of k-nearest neighbors in feature space.

Preliminary experiments on CIFAR-10 and CIFAR-100 for classification, and BSD68 and CBSD68 for denoising, indicate competitive performance with standard convolution baselines. Integration into VGG architectures demonstrate that ConvNN can effectively replace standard convolution operations while maintaining comparable accuracy.

This reinterpretation of convolutions provides a new perspective for designing hybrid architectures that interpolates between local and global processing, potentially combining convolution's spatial locality bias with ConvNN's ability to capture global feature relationships.

## Feature Work and Connection to Attention

The k-nearest neighbor interpretation of convolution suggests connections to other architectural components. Self-attention mechanisms compute similarities between all positions using query-key dot products, effectively selecting neighbors based on feature similarity rather than spatial proximity. The softmax normalization in attention performs soft selection where all positions contribute with weighted values summing to one, while convolution performs hard selection of a fixed spatial neighborhood. ConvNN's hard neighbor selection contrasts with attention's soft selection, providing interpretable neighbor assignments and potentially improved computational efficiency.

Despite extensive empirical comparisons between CNNs and Transformers, the literature lacks a thorough theoretical analysis of their fundamental similarities and differences. Both operations aggregate information from multiple positions, yet employ different mechanisms. Convolutions used fixed spatial kernels with learned weights, while attention computes dynamic weights based on content similarity. Understanding these operations through a unified framework with k-nearest neighbors as the bridge that could leverage the strengths of both mechanisms. Future work will address this gap by demonstrating that both convolution and attention can be understood as specific instances of a more general neighbor aggregation framework, where the key distinction lies in neighbor selection rather than in the aggregation mechanism itself.

## References


Chen, T.; et al. 2020. A simple framework for contrastive learning of visual representations. In *ICML*.

He, K.; et al. 2020. Momentum contrast for unsupervised visual representation learning. In *CVPR*.

Plötz, T.; and Roth, S. 2018. Neural nearest neighbors networks. In *NeurIPS*.

Qi, C.; et al. 2017. PointNet: Deep learning on point sets for 3D classification and segmentation. In *CVPR*.

Shi, W.; et al. 2016. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *CVPR*.

Wu, Z.; et al. 2018. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*.