

A Supplementary Material

A.1 Experimental Setup

We build on the official [MOMENT repository](#) and the PyTorch implementation of [PointNet](#) and [PointNet++](#), re-writing kernels in C++/CUDA to support radius-ball neighbourhood sampling with a fixed radius of 200m. All experiments run on a single NVIDIA **H200** (80 GB) GPU under PyTorch 2.3.

Unless stated otherwise we keep the default MOMENT configuration (`moment-base`, forecast horizon = 60), freeze the encoder as in the official fine-tuning recipe, and train only the PointNet block and forecasting head. We initialise all backbone weights from the publicly released moment-base checkpoint; the new projection layer and the PointNet MLP use Xavier initialisation. PointNet uses a three-layer MLP (64-128-256) with ReLU and BN, followed by global max pooling; its output ($\mathbf{s}_i \in \mathbb{R}^{256}$) is concatenated to the padded displacement history, giving a 512-d vector per scatterer. Batch size = 2048, Adam with $\text{LR} = 10^{-4}$, One Cycle schedule (30 epochs, warmup 30%), and mixed precision.