Mines Paris — PSL
Intelligence Artificielle, Systèmes et Données (IASD)
Nuages de Points et Modélisation 3D
Students: Alexandros Kouvatseas & Luka Lafaye de Micheaux

Theory Question: What is happening for the 2D spiral input point cloud?

For the 2D spiral input, each pixel is fed through the network and mapped to a logit (using tanh in the last layer) that lies between -1 and 1. In this setup, if the logit is below 0 the pixel is classified as orange, and if it is 0 or above it is classified as blue. This creates a decision boundary that divides the spiral data into two classes.

Question 1: Give your test accuracy on ModelNet10_PLY or ModelNet40_PLY with your MLP neural network. Give your parameters (learning rate, nb epochs). Comment the results.

MLP model	
Parameter	**Value**
Learning Rate	0.005
Number of Epochs	30
Test Accuracy on ModelNet10_PLY	19.1 %

Table 1: Hyperparameters and test accuracy of the MLP model.

The accuracy of 19.1% is only slightly above random chance (10% for 10 classes), indicating that the MLP is barely learning discriminative features from the data. This is primarily due to the flattening of the input point cloud (1024 points with 3 coordinates each $\rightarrow$ 3072 features). This process discards spatial relationships and local geometric structures, which are crucial for understanding 3D objects.

ModelNet10_PLY contains complex 3D CAD models, and a simple MLP is likely too limited to capture the inherent spatial and hierarchical features present in the point clouds. The training logs show that although the loss decreases over epochs (eventually reaching very small values), the test accuracy remains low. This suggests that the model is either underfitting the complexity of the data or failing to generalize well.

In summary, the poor performance highlights the need for more sophisticated architectures, such as PointNet, which can better leverage the spatial structure of 3D point clouds.

Question 2: Give your test accuracy with the basic version of PointNet on ModelNet10_PLY or ModelNet40_PLY. Give your parameters (learning rate, nb epochs). Comment the results.

Basic PointNet	
Parameter	**Value**
Learning Rate	0.005
Number of Epochs	30
Test Accuracy on ModelNet10_PLY	87.3%

Table 2: Hyperparameters and test accuracy of the Basic PointNet model.

With the same parameters used for training the MLP, the basic PointNet model (without T-Net) achieves accuracies ranging from 85% to 90% on ModelNet10_PLY. This result represents a significant improvement compared to the simple MLP. PointNet applies shared 1D convolutional layers to each point, preserving local relationships. Then, a symmetric max-pooling operation efficiently aggregates the local features into a robust global representation that is invariant to the ordering of the points. The hierarchical structure, which first extracts local features and then combines them into a global feature, enables the network to better capture the complexity and variability of 3D objects.

Question 3: Give your test accuracy with PointNetFull adding the 3*3 T-Net on ModelNet10_PLY or ModelNet40_PLY. Give your parameters (learning rate, nb epochs). Comment the results and compare to the basic version results.

Full PointNet with TNET	
Parameter	**Value**
Learning Rate	0.005
Number of Epochs	60
Test Accuracy on ModelNet10_PLY	90.3%

Table 3: Hyperparameters and test accuracy of the Fully implemented PointNet model.

With the implementation of the complete PointNet architecture, including T-Net, we observe slightly better results compared to the version without T-Net. However, achieving stability requires 60 epochs instead of 30. This indicates that while T-Net improves the model's ability to learn transformation-invariant features, it also introduces additional complexity, making the optimization process slower. The network takes longer to converge as it learns both the classification task and the optimal transformations for alignment. Nonetheless, the improved performance suggests that the added training time is justified by the enhanced robustness and generalization of the model.

Question 4: Find a new data augmentation on 3D point clouds. Explain your idea. Give your test accuracy with and without your data augmentation on ModelNet10_PLY or ModelNet40_PLY (using basic or full version of PointNet). Give your parameters (learning rate, nb epochs). Comment your results.

In this part we tested two different methods, random scaling and random shreding. Random scaling involves applying a uniform scaling factor to the entire point cloud, effectively resizing the object while maintaining its proportions. In contrast, random Shearing introduces distortions along the x and y axes by applying a shear transformation matrix, altering the shape of the object while preserving its overall structure.

By observing *table 4*, we see that the data augmentation methods have not had any impact to the full architecture of the PointNet. This means, that the fact that it contains T-net, the model already learns an affine transformation matrix that aligns point clouds to a canonical space and it effectively normalizes variations in the input data. As a result, external augmentation techniques like random scaling and Shearing do not provide additional benefits, as the network already adapts to these transformations internally. This reinforces the idea that T-Net enhances invariance to geometric transformations, making explicit augmentation less influential in improving model performance.

Full PointNet with TNET		
Parameter	**Random Shearing**	**Random Scaling**
Learning Rate	0.005	0.005
Number of Epochs	60	60
Test Accuracy on ModelNet10_PLY	91.0%	89.8%

Table 4: Hyperparameters and test accuracy of the Fully implemented PointNet model with different augmentation techniques.

On the other hand, using Basic PointNet, as observed in *table 5*, there is an improvement in performance when applying random Shearing. Additionally, by examining the training logs, we can see a faster convergence compared to the baseline. In contrast, random scaling leads to a slight decline in overall performance, likely due to distortions affecting the spatial relationships between points. Convergence happens slowly, and there are various ups and downs during the training of the model.

Even though the difference in performance is not substantial, in more complex point clouds, such augmentation techniques could play a more significant role. In scenarios where objects exhibit high intra-class variability or are presented at different scales, carefully selected augmentations might help the model generalize better. The observed improvements with random Shearing suggest that mild deformations can help introduce beneficial variations into the training set, making the model more robust to real-world variations.

Basic PointNet		
Parameter	**Random Sampling**	**Random Shearing**
Learning Rate	0.005	0.005
Number of Epochs	30	30
Test Accuracy on ModelNet10_PLY	85.2%	90.2%

Table 5: Hyperparameters and test accuracy of the Basic PointNet model with different augmentation techniques.