Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Classifying Synthetic 3D Particle Shapes Using Classical Descriptors and Point Cloud Networks

Tanisha Kumari
Indian Institute of Technology Roorkee
Mayank Goswami
mayankfph@iitr.ac.in

Indian Institute of Technology Roorkee

Method Article

Keywords: particle morphology, point clouds, shape descriptors, PointNet, PointNet++, synthetic data

Posted Date: November 9th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11643386/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

Classifying Synthetic 3D Particle Shapes Using Classical Descriptors and Point Cloud Networks

Tanisha Kumari[1] and Mayank Goswami [1, 2, #]

[1]Department of Physics, IIT Roorkee
[2]*Mehta Family School of Data Science and Artificial Intelligence*, IIT Roorkee,
Roorkee, Uttarakhand-247667, India.
[#]mayank.goswami@ph.iitr.ac.in

Abstract—This paper evaluates geometric rules, Random Forest, PointNet and PointNet++ in the classification of synthetic particles as either spherical, angular, elongated or platy. We created 8,000 labeled particles and a further 800 elongated/platy boundary set. Random Forest achieved 100% accuracy on the original test split while both deep 2,048-point network architectures reached 99.92%. On the boundary set, the validation tuned rule achieved 100% accuracy and Random Forest achieved 80.12%. Of the eight complete deep architectures tested, PointNet++ with 2,048 points and without any augmented training rotations achieved the highest accuracy of 87.88%. A sensitivity analysis was carried out on factors including number of points, training augmentations, new test rotations, rule thresholds and class recall. The uncertainty estimates utilized bootstrapping within geometry groups. Interpretation is not straightforward: of 1,200 original test particles, 916 have exact geometric matching within training. Random Forest also achieved 100% accuracy on a further geometry-based test set, while the deep learning models were not tested on this split. Streamlit provides the inference of saved models.

Index Terms— particle morphology; point clouds; shape descriptors; PointNet; PointNet++; synthetic data

I. INTRODUCTION

Shape classification is useful in the study of powders and aggregates because morphology influences material behaviour. A classifier could support quality checks, but its usefulness depends on how well its labels and test data reflect the particles it will encounter. We therefore compare transparent shape measurements with two networks that learn directly from point clouds. The question is practical: for this controlled dataset, is a more complex model justified by better classification or robustness near a class boundary?

All the four classes in the current project have been categorized on the basis of geometrical generators. The spheres are for compact particles, while the superellipsoid blocks are for angular particles. The ellipsoids with one elongated or short axis are for elongated and platy particles respectively. The perfect geometrical generators provide exact categorization labels without needing any manual labeling. This is because of their simplicity compared to fractured or scanned particles.

II. PREVIOUS WORK AND THEORY

The PointNet model operates on the unordered point cloud with shared layers and a symmetric pooling function, hence the prediction of this model is independent of the order of input points [1]. The PointNet++ introduces the concept of neighbourhoods at different scales [2]. These two models were built for 3D point cloud learning in general and the results presented in literature are not guaranteed to work better than simple models in our particle data set.

In our classical method we used the sphericity and axis ratios computed using the constructed geometry of the particles. The sphericity is the ratio between the surface area of the particle and that of the sphere which has the same volume. The principal component directions characterise the dispersion of the surface points along the three axes. These measures are understandable and they are not able to describe every aspect of the local surface geometry.

III. MATERIALS AND METHOD

A. Dataset

The primary dataset comprises 2,000 particles per class, making up 8,000 particles overall. Its initial partitioning consists of 5,600 training, 1,200 validation, and 1,200 test particles. The secondary boundary dataset includes 400 elongated and 400 platy particles around generator ratio thresholds. It has not been included in model training and evaluation. Each generated particle is represented by a $64 \times 64 \times 64$ binary voxel field and a 2,048 point cloud. Point clouds are centered, rescaled to a unit sphere and rotated at random. The data generator has seed 42.

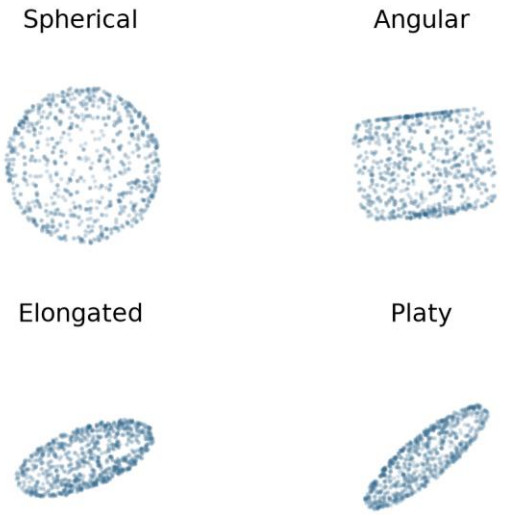


Fig. 1. Example surface point clouds from the four generated classes.

B. Classical Models

We measure sphericity from a triangular surface mesh and three ratios from ordered principal-component standard deviations: major/middle elongation, middle/minor flatness, and major/minor aspect ratio. The fixed rule first tests elongation, then flatness, then sphericity. A second rule searches thresholds using validation macro-F1 only. A Random Forest with 300 trees learns from the same four measured descriptors. Its scaling step and forest are fitted on training data only. Neither test nor boundary labels are used to select thresholds or weights.

C. Point-Cloud Models

The local implementation of PointNet utilizes shared pointwise layers, learning transformations, max pooling, and 4 classes. The PointNet++ implementation includes hierarchical neighborhood aggregation. Both implementations use Adam optimizer, batch size 32, cross entropy loss function with 0.1 label smoothing, and macro-F1 for validation on held-out set to save the best weights. The training is allowed to run for a maximum of 150 epochs, but with early stopping if there is no improvement in validation results for 20 epochs.

We tested also 512- and 1,024-points implementations as well as 2,048-points implementations without additional training rotation.

D. Evaluation and Geometry Audit

Reported on the original test set are accuracy and four-class macro-F1. The boundary set has only two true classes; the shown F1 is an average of the two existing classes. Newly rotated test sets serve as additional sensitivity tests. Voxels were hashed by geometry to detect duplicate geometries in all splits. An additional split by geometry puts each unique voxel geometry into only one of the main splits within that class. The classical models have been tested using this geometry split protocol. Two deep learning grouped runs have been terminated before evaluation.

E. Sensitivity Analysis

Our analysis included eight deep configurations that were completed: each architecture with 512, 1,024 and 2,048 points, including extra training rotations, as well as one configuration with 2,048 points without extra rotations. We analyzed the validation threshold grid and predictions after each test particle was rotated once by a new random rotation. No extra rotation runs use the initially rotated point clouds from the dataset. These are sensitivity analyses of the stored runs, where no training of models occurred. Sensitivity is class recall, TP/(TP+FN).

The predictions were aligned by the sample identifier and resampled by 2,000 paired bootstrap samples of entire voxel geometry classes in each true class while maintaining the original proportions (seed 20261029). There are 414 different geometries in the original test and 38 geometries in the boundary set. Random resampling of exactly repeated geometries takes into account some dependencies but not similarities among geometries or training seeds variance. The confidence intervals are exploratory, there is no correction for multiple comparisons. An interval could collapse to 100% if all observed predictions were correct, which does not guarantee zero error.

IV. RESULTS

TABLE I
Held-out results on the original split. All four classes are present.

Method	Accuracy	Macro-F1
Fixed rule	95.83%	0.9584
Validation-tuned rule	99.92%	0.9992
Random Forest	100.00%	1.0000
PointNet, 2,048 points	99.92%	0.9992
PointNet++, 2,048 points	99.92%	0.9992

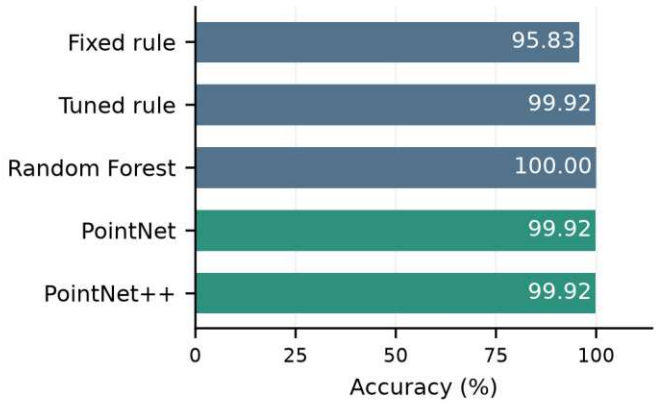


Fig. 2. Original test accuracy of the principal methods.

The original test scores are close to perfect for most methods. The boundary set gives a different ranking:

TABLE II
Boundary results. Only elongated and platy particles have true labels in this set.

Method	Accuracy	Present-class F1
Fixed rule	0.25%	0.0050
Validation-tuned rule	100.00%	1.0000
Random Forest	80.12%	0.8773
PointNet, 2,048 points	52.62%	0.5529
PointNet++, 2,048 points	62.62%	0.7053

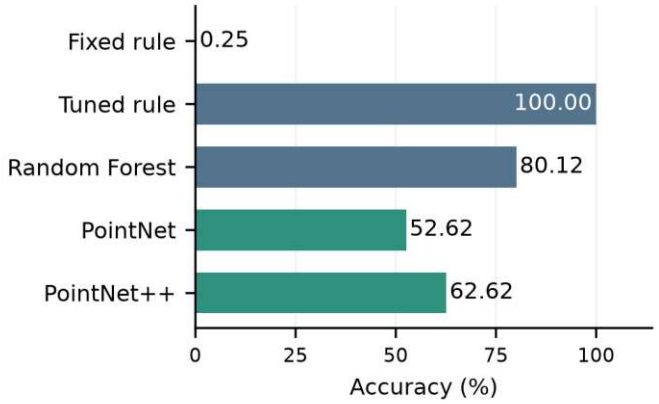


Fig. 3. Boundary accuracy. The validation-tuned rule performs best on this particular constructed set.

There were 2,075 unique voxel geometries among the 8,000 main samples discovered by the geometry analysis. In the first split, there were 916 out of 1,200 test samples which had the same voxel geometry as the training dataset. Geometry-based splitting ensures that this exact match is not present in its training, validation, and test splits. For the test set of the geometry grouping method, fixed-rule accuracy is 96.28%, tuned-rule accuracy is 68.34% and Random Forest accuracy is 100%. The test samples of the geometry group method are different from those of the first test set.

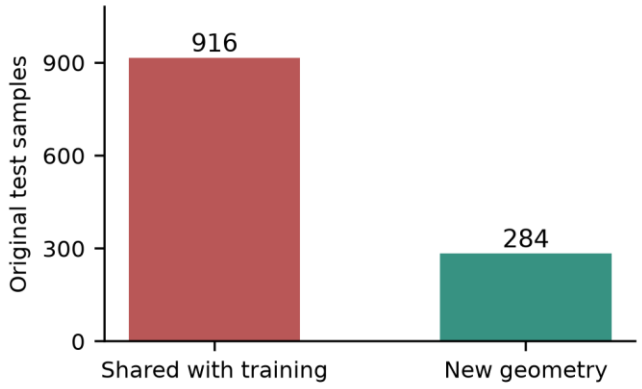


Fig. 4. Most original test samples have an exact voxel match in training.

A. Sensitivity of Model Performance

More points do not consistently improve boundary performance. With extra training rotations, PointNet++ boundary accuracy is 84.75%, 83.50% and 62.62% at 512, 1,024 and 2,048 points. At 2,048 points, the run without extra rotations exceeds the rotation-augmented run by 25.25 percentage points (95% geometry-bootstrap interval 22.52 to 28.30). Each configuration was trained once, so the observed difference cannot isolate augmentation from training-run variation.

TABLE III
Completed deep runs and sensitivity to point count, training augmentation, and fresh test rotations. Each configuration has one training run.

Model and configuration	Test (%)	Boundary (%)	Fresh rot. boundary (%)
PointNet 512 extra rotations	100.00	72.38	66.62
PointNet 1,024 extra rotations	99.75	74.75	76.75
PointNet 2,048 extra rotations	99.92	52.62	51.38
PointNet 2,048 no extra rotations	99.92	58.25	59.38
PointNet++ 512 extra rotations	99.92	84.75	80.50
PointNet++ 1,024 extra rotations	100.00	83.50	84.38
PointNet++ 2,048 extra rotations	99.92	62.62	64.62
PointNet++ 2,048 no extra rotations	99.92	87.88	87.50

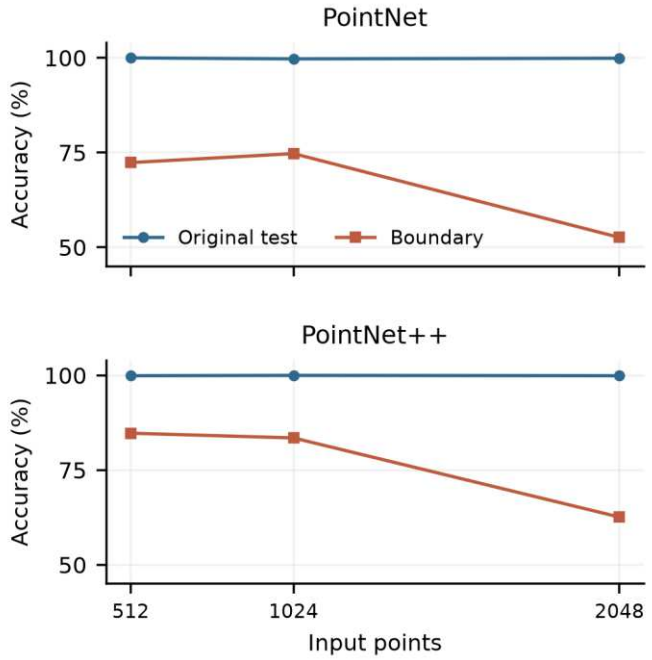


Fig. 5. Original-test scores remain near 100% while boundary performance changes with the point count. Lines connect separately trained models.

After just one additional random rotation per boundary particle, PointNet++ with 2,048 points and without additional training rotations obtains an accuracy of 87.50%, corresponding to a difference of −0.38 percentage points (95% CI [−3.32 to 2.27]). However, 12.63% of its label predictions changed. Thus, while the overall accuracy is almost the same, individual predictions are not consistent. One additional random rotation per particle was considered.

In the case of boundary particles, Random Forest recall on elongated and platy particles is 99.00% and 61.25%, respectively. The best deep boundary run that we have seen achieves 97.75% elongated and 78.00% platy recalls. Tuning the rule gets 100% for both types. There are no spherical or angular true particles in the boundary set, so these numbers cannot be used to measure sensitivity. Validation sets were used for the construction of thresholds.

Random Forest achieves 100% on both the original test set and a new classical geometrically grouped test set, and thus can be considered as a strong learned baseline for the artificial task. The validated rule works best on the artificial boundary set with 100% accuracy. Out of all the finished deep learning trials, PointNet++ with 2,048 points and without any additional rotations during training has the largest boundary accuracy of 87.88%. The advantage of PointNet++ relative to Random Forest on the same artificial boundary set is 7.75 percentage points, with a bootstrap interval of 0.56 to 15.11 percentage points. This comparison was done after analyzing the results and requires further confirmation.

B. Streamlit Demonstration

An example of point cloud model checkpoint can be seen through the Streamlit interface. It presents the chosen checkpoint with accuracy in test/boundary mode, predicted class, softmax value, and the three-dimensional visualization of point cloud. According to the recording, in PointNet++, when there were 2,048 points without rotation, spherical_01700 was correctly recognized as spherical with the value of 88.1%. Original-test accuracy is 99.9%, and boundary accuracy is 87.9% after rounding. This information represents the actual evaluation results, and this particular example is only an illustration of inference process, which does not represent an additional accuracy measure, calibration study, usability study, or real-particle validation..

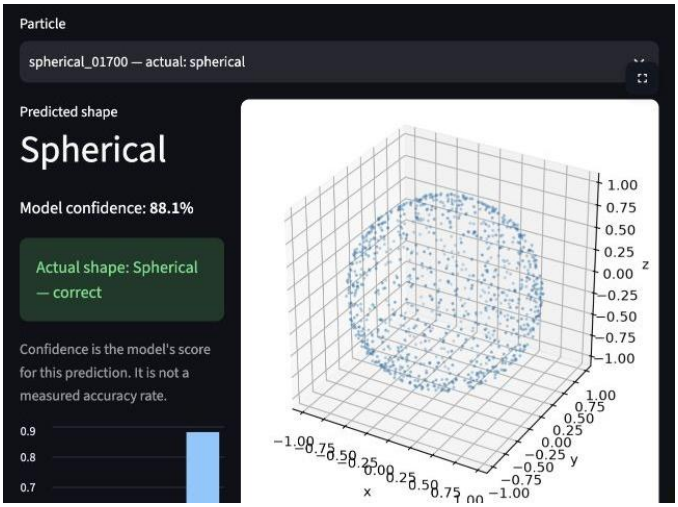


Fig. 6. Actual Streamlit prediction for the labeled test particle spherical_01700. The model score is 88.1%; it is not a measured accuracy for this individual input.

V. DISCUSSION

With four descriptors already measured, the generated classes are separable enough by their values for the Random Forest to deal with this synthetic problem without a point cloud network. As a result of the rule comparison, it follows that even a simple approach may be fragile because it is unable to separate the selected boundary set with fixed thresholds, whereas the same set is separated by perfectly tuned thresholds. It does not imply that the tuned rule will be perfect in separating any other boundary sets since the thresholds calculated from the generator parameters and sample measurements do not have to match each other.

The original split allows making an optimistic conclusion about the generalisation ability of the models because most of the time they are tested on new surface samples or orientations of geometry samples from the training set. Split into groups of exact geometries is a more serious test, but there are still similar geometries and generator families present in all splits.

The dataset is artificial, well-balanced, and intentionally kept simple. No microscopes, no CT-scans, no irregular fragments, no scanner noise, and no artifacts particular to a material are involved. A model score obtained using this dataset cannot be taken as a calibrated probability or indicator of the model's preparedness for industrial decision-making tasks. The future work should involve the actual materials along with independently validated labels, splits according to their origin or family of shapes, and assessment of missing points and segmentation mistakes.

VI. CONCLUSION

This project provides a functional pipeline of synthetic 3D particle classification, together with an interactive demo on Streamlit platform. Random Forest provides a good learned baseline in the original and geometry group classical tests. For the created boundaries dataset, the optimal rule provides the best results; meanwhile, PointNet++ with 2,048 points and without additional rotation provides the best result for this dataset as the most optimal deep run. Sensitivity to point number, rotation and dataset creation process makes it impossible to rank all models for all possible cases.

DATA AND CODE AVAILABILITY

The above mentioned submission includes the following: source code in Python, dependencies, weights for trained models, split information, results of evaluation along with sensitivity analysis tables and figures, as well as generated dataset divided into training, validation, testing, and boundary folders.

ETHICAL CONSIDERATIONS

This dataset is not a representation of all particle materials, sizes, imaging techniques, and surface defects. The application of this classifier for quality control without the validation process on the actual material may lead to false decisions. The evaluation process must provide the class-level errors and

performance per material and data acquisition source. Uncertain cases must undergo human review.

RESOURCES:

Video: https://youtu.be/uP0_tV2A7t0

Code:

REFERENCES

[1] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2017. [Online]. Available: https://arxiv.org/abs/1612.00593

[2] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space," in Advances in Neural Information Processing Systems, vol. 30, 2017. [Online]. Available: https://arxiv.org/abs/1706.02413