Dual-Stream Semantic Segmentation Architecture for Point Cloud Analysis

Brendon Hales, Philly Tang
Dignitas Technologies
Orlando, Florida
bhales@dignitastechnologies.com,
ptang@dignitastechnologies.com

Troy Crawford, Marjaneh Safaei
Dignitas Technologies
Orlando, Florida
tcrawford@dignitastechnologies.com,
msafaei@dignitastechnologies.com

ABSTRACT

Semantic segmentation, a crucial task in computer vision, has evolved to encompass point cloud data alongside traditional two-dimensional images. In this paper, a novel approach is presented to perform semantic segmentation on point cloud data, leveraging two heterogenous deep neural network streams, trained over two distinct data modalities: the first stream processes two-dimensional images, while the second stream focuses on three-dimensional point cloud data. By integrating these streams, the proposed method exploits the complementary information inherent in each data modality, enhancing the segmentation accuracy and robustness. Here, an ensemble methodology is proposed by employing stacking algorithm to consolidate the outputs from both streams, enabling the final decision-making process. Ensemble methods are machine learning techniques that combine the predictions of multiple individual models to produce a more accurate and robust prediction. In stacking, multiple classification or regression models are combined using a Meta Classifier that is trained on the outputs of the base-level models as features to deliver the final classification results. Meta Classifier is a learning algorithm that learn from other learning algorithms. The proposed ensemble method learns a Meta Classifier by comprising of 2D and 3D semantic segmentation deep neural networks, the two heterogeneous base classifiers. This ensemble framework combines the strengths of individual models, effectively mitigating errors and improving overall semantic segmentation performance. Experimental results on benchmark datasets demonstrate the effectiveness of the proposed approach, achieving superior segmentation accuracy compared to the state of the art. Point cloud analysis has various applications, including autonomous driving, robotics, and augmented reality, where accurate and efficient segmentation of point cloud data is essential for scene understanding and decision-making.

ABOUT THE AUTHORS

Brendon Hales is a Software Engineer at Dignitas Technologies who is primarily focused on machine learning and software integration. He has worked on a variety of machine learning tasks such as semantic segmentation of 2D images and 3D point cloud space, and computer vision problems such as object identification, and real-time object tracking. He has been vital in coordinating integration of the current machine learning work with other projects at Dignitas Technologies. He currently holds a B.S. in Computer Engineering from the University of Central Florida.

Philly Tang is a Software Engineer at Dignitas Technologies. His primary focus is on software development, software integration, and machine learning specifically in semantic segmentation, instance segmentation, and object detection in real time. His contributions have attributed to resolving test cases and ensuring the software are tested, integrated, and complied to the company's policy before being sent for delivery. He holds a B.S in Geomatics at University of Florida and has plans on pursuing his master's in computer science at University of Central Florida.

Troy Crawford is a Software Engineer at Dignitas Technologies. His primary focus is on machine learning software development and integration with computer vision problems such as semantic segmentation of 2D images and 3D point clouds, object detection, human gesture recognition and real-time deep learning solutions. His efforts of software integration are by evaluating testing methods and measures to ensure quality, packaging and documenting according to company policy and procedures. He holds a B.S. in Computer Science at the University of Central Florida.

Marjaneh Safaei is a senior Machine Learning research scientist at Dignitas Technologies**.** Dr. Safaei has a strong research background and a Ph.D. in Computer Science from University of Central Florida. She will leverage her 9 years of experience in designing supervised and unsupervised ML and Deep Learning algorithms, Computer Vision, and Image Processing methodologies to augment our ML solution. She has published several scientific conference papers in top-tier artificial intelligence and ML conferences. She received multiple awards and scholarships from prestigious conferences, such as IEEE and AAAI. She also serves as a reviewer in various IEEE transactions.

Dual-Stream Semantic Segmentation Architecture for Point Cloud Analysis

Brendon Hales, Philly Tang
Dignitas Technologies
Orlando, Florida
bhales@dignitastechnologies.com,
ptang@dignitastechnologies.com

Troy Crawford, Marjaneh Safaei
Dignitas Technologies
Orlando, Florida
tcrawford@dignitastechnologies.com,
msafaei@dignitastechnologies.com

1. INTRODUCTION

Semantic segmentation is a challenging task in computer vision that involves classifying each pixel in an image or each point in a point cloud into predefined categories. While 2D semantic segmentation techniques have been widely studied and applied, 3D semantic segmentation methods are gaining attention due to the increasing availability of 3D data from sensors that have decreased in cost and complexity of data retrieval. In this paper, we will explore an innovative method that combines the strengths of both 2D and 3D semantic segmentation techniques to achieve more accurate and robust results for point cloud data. The stacking ensemble method involves combining the predictions of multiple base models to create a more accurate and reliable final prediction. By stacking 2D and 3D semantic segmentation networks in an ensemble, this method aims to leverage the strengths of both approaches and address their individual limitations in segmenting point cloud data effectively.

The proposed end-to-end semantic segmentation framework leverages three main advantages of employing an ensemble methodology; **Enhanced feature representation:** The two-stream proposed method can leverage the diverse and complementary feature representations learned by each network. The 2D network excels at capturing contextual information from 2D projections by focusing on the relationship of the nearby pixels, while the 3D network is adept at capturing spatial information in the point cloud data in 3D space. By combining these feature representations, the ensemble method can achieve a more comprehensive understanding of the scene and improve segmentation accuracy. **Improved generalization and robustness:** Ensemble methods are known for their ability to reduce overfitting and improve generalization by combining the predictions of multiple models. By stacking 2D and 3D semantic segmentation networks in an ensemble, the method can enhance the generalization capabilities of the models and improve their robustness to variations in the input data. This can lead to more consistent and reliable segmentation results across different scenes and datasets. **Adaptive fusion of predictions:** The stacking ensemble method allows for the adaptive fusion of predictions from the base 2D and 3D semantic segmentation networks. By learning how to combine the outputs of the individual networks based on the characteristics of the input data, the ensemble method can adaptively adjust the fusion strategy to optimize segmentation performance. This dynamic fusion of predictions can lead to more accurate and context-aware segmentation results in diverse point cloud scenes. In complex urban environments with various objects such as cars, pedestrians, and buildings, the proposed stacking ensemble method combines the pixel-wise prediction of a 2D semantic segmentation network trained over RGB images, and the point-level prediction of a 3D semantic segmentation network trained over raw point cloud data. By leveraging the contextual understanding of the 2D network and the spatial information of the 3D network, the ensemble method accurately segments each object in the scene and captures their spatial relationships with higher precision. Moreover, combining 2D semantic segmentation with 3D networks can facilitate the transfer of knowledge and features between tasks and domains, enabling the model to leverage pre-trained models, transfer learning techniques, and domain adaptation strategies. This can lead to more efficient training, faster convergence, and improved performance on tasks with limited training data or domain shifts. By capitalizing on the strengths of both 2D and 3D representations, the hybrid approach can achieve state-of-the-art results in semantic segmentation tasks and advance the capabilities of computer vision systems.

The remainder of this paper is structured as follows: Section 2, briefly describes different approaches currently used for point cloud semantic segmentation in the literature. Section 3 represents the technical approach for the ensemble dual-stream semantic segmentation framework in detail. Section 4 illustrates extensive experimental results, benchmarking the ensemble semantic segmentation model over three different datasets. Finally, paper conclusion outlining the significance of the ensemble approach along with the reasons supporting its importance is presented in Section 5.

2. RELATED WORK

Point clouds contain valuable geometric data for each point, but the lack of inherent structure makes it challenging to determine local context. Various methods for segmenting point clouds are recently explored in the computer vision community, from preprocessing to alternative representations, to direct processing of the raw data. Grid-based techniques, inspired by image processing, often convert point clouds into grid-based representations to leverage convolutional neural network operations. These representations include two-dimensional pixel images (Su, Maji, 2015) (Wu, Yang, and Wang, 2019), a bird's eye view of the scene (Zhang, Luo, 2018) (Fei, Peng, 2021), or a three-dimensional voxel grid (Wu, Song, 2014) (Maturana, Scherer, 2015). However, relying on the grid-based representation could be challenging due to the large section of empty space in grid representations which leads to the increased computation complexity, inaccurate classification performance and memory inefficiencies. Empty spaces in grid-based semantic segmentation can arise from the nature of the image data, the grid structure, and the capabilities of the segmentation model or algorithm. Addressing these challenges requires careful consideration of these factors and the implementation of strategies to effectively handle empty spaces in the segmentation process. Point-based methods, like PointNet (Qi, Su, 2016), operate directly on raw point cloud data, utilizing Multi-Layer Perceptrons, MLP, for individual point processing. More research rapidly followed, extending it directly such as PointNetLK (Aoki, Goforth, 2019) and PointNet++ (Qi, Yi., 2017), or developing new algorithm using MLPs as a base. Other approaches, such as RandLA-Net, leverage MLPs and K-Nearest Neighbors to capture local relationships between points. One of the key differences between point-based and grid-based semantic segmentation lies in their handling of empty spaces. In grid-based segmentation, empty spaces can pose challenges, as discussed earlier, due to the disruption of the grid structure and potential mislabeling of pixels. Point-based segmentation, on the other hand, may not be as affected by empty spaces since it treats each pixel independently.

Some other works employ Coarse-to-fine processing strategies to extract features from raw point clouds, especially when local context is lacking (Lu, Wang, 2021). Some networks first identify distinct objects before classifying points, while others focus on extracting fine features after down-sampling the point cloud. These approaches all assume that fine features exist when extracting and propagating them, which does not hold when a scan's density is inhomogeneous. Dealing with density variation in point clouds is crucial, as local densities can vary significantly. However, most of the existing approaches do not address density variation across the entire network but they do take steps to limit the effect on individual network layers (Li, Wang, 2020). Alternative point cloud representations such as voxels tackle density by either weighting each voxel based on how many points it has (Hu, Kuai, 2022), or by implementing a minimum density floor, ignoring sparse sections entirely (Chen, Chen, Xu, 2021). Therefore, the segmentation of point clouds involves a range of techniques to handle the unique challenges posed by the lack of inherent structure and density variations in the data.

Point cloud data, which consists of a collection of points in 3D space, poses unique challenges for semantic segmentation, such as irregular point densities, varying point distributions, and complex spatial relationships. Ensemble methods offer a promising solution to address these challenges and enhance the accuracy and robustness of segmentation results. One common ensemble approach for semantic segmentation in point clouds is the use of multiple neural networks with different architectures or initializations. By training and combining multiple networks, each with its strengths and weaknesses, ensemble models can leverage diverse representations and learn complementary features to improve segmentation accuracy. For example, combining a convolutional neural network (CNN) with a graph neural network (GNN) can capture both local and global spatial dependencies in point cloud data, leading to more accurate segmentation results. Another ensemble strategy is to integrate multiple segmentation algorithms or techniques, such as region-based methods, graph-based methods, or clustering algorithms. By combining the outputs of different segmentation approaches, ensemble models can benefit from the strengths of each method and mitigate their individual limitations. For instance, combining region growing with graph cuts or clustering with convolutional neural networks can effectively segment objects in point clouds with varying structures and densities.

The proposed ensemble approach in this paper, integrates 2D semantic segmentation techniques with 3D networks, leveraging the strengths of both domains and enables more comprehensive and accurate analysis of spatial data. One key differentiator of combining 2D semantic segmentation with 3D networks is the ability to exploit the rich contextual information provided by 2D images in conjunction with the depth and spatial relationships captured by 3D data. While 2D semantic segmentation excels at segmenting objects and regions in images based on pixel-level features, 3D networks are specialized in processing volumetric data and capturing geometric structures and spatial layouts. Furthermore, the integration of 2D and 3D data enables multi-modal learning and feature fusion, allowing the model

to learn complementary representations from different sources and modalities. This fusion of information across modalities enhances the model's ability to generalize to diverse scenarios and adapt to variations in data distribution.

3. TECHNICAL APPROACH

Semantic segmentation plays a crucial role in various applications, such as remote sensing, medical image analysis and autonomous driving. However, achieving accurate segmentation results can be challenging due to the complexity and variability of real-world data. In recent years, deep learning models have shown promising results in semantic segmentation tasks (Boulch, Guerry, 2018) (Tchapmi, Choy, 2017) (Qi, Su, 2016). By leveraging the power of deep learning, high-level features from images or 3D volumes are extracted and make more accurate predictions.

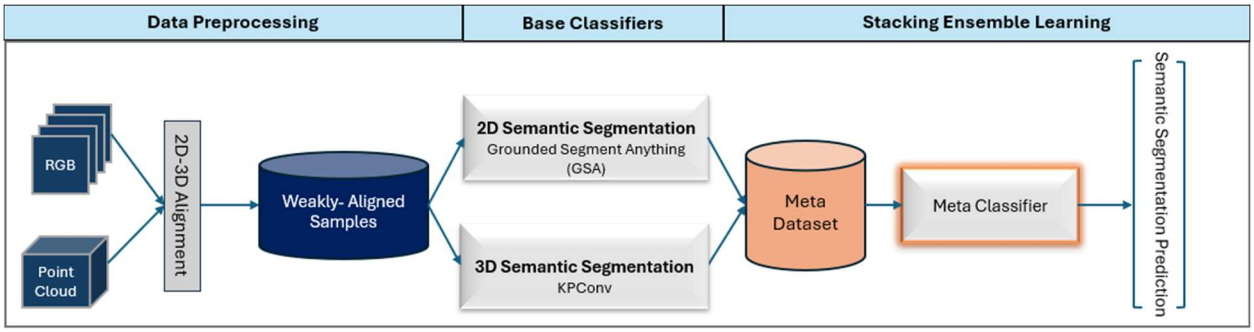

Figure 1. Schematic overview of the two-stream ensemble semantic segmentation framework

One way to improve the segmentation performance is to use ensemble learning, which combines multiple models to make the final prediction. Stacking is a popular ensemble technique that involves training a Meta learner, a learning algorithm that learn from other learning algorithms, to combine the predictions of base classifiers. In the proposed two-stream ensemble framework, the base classifiers, two heterogeneous semantics segmentation networks in 2D and 3D domain, are fused to improve the semantic segmentation performance. A 2D Convolutional Neural Network (CNN) for image segmentation is well-suited for capturing features in 2D images, while a 3D CNN can handle the additional depth information in 3D volumes for volumetric segmentation. By combining the predictions of these two models using a Meta learner, we can leverage the complementary strengths of both models. Furthermore, stacking can help to reduce overfitting and improve generalization. By training multiple base classifiers on different subsets of the data or using different architectures, we can reduce the risk of overfitting to a specific dataset or model. The Meta learner can learn to combine the predictions of base classifiers in a way that maximizes the overall segmentation performance on unseen data. Figure 1 presents the schematic overview of the hybrid semantic segmentation framework. First, weakly-aligned samples are generated within the data preprocessing steps. Next, the weakly-aligned samples serve as the input for the two base classifiers, to generate the prediction matrices. Finally, Meta Classifier is trained over the Meta dataset, the output of the base classifiers.

3.1 2D Semantic Segmentation

When it comes to implementing a suitable 2D semantics Segmentation framework, the ideal framework should be able to detect objects and draw the segmentation mask for that object. To build the first base classifier in the proposed ensemble two-stream pipeline, Detic (Zhou, Girdhar, 2022), an objects detection and semantic segmentation framework, developed by Facebook, was first utilized to perform semantic segmentation over 2D RGB images. Detic trains classifiers over 2D data and expands the vocabulary of detectors to tens of thousands of concepts. However, when employing a custom vocabulary, Detic occasionally struggles to capture the entire contextual meaning and instead focuses on individual words from the sentence. This limitation may be attributed to the fact that Detic extracts image labels from captions using a simplistic text-match approach, primarily trained on singular words. Therefore, Grounded Segment Anything (Ren, Mintun, 2024) an open-source semantic segmentation framework is next employed to serve as the first base classifier in the ensemble architecture. Grounded Segment Anything (GSA) utilizes the concept of text prompt and segmentation to create masks for specific object in images. GSA uses two pretrained models: GDINO and SAM. GDINO (Grounding DETR with Improved DeNoising Anchor Boxes) is an object detection model that can identify various objects based on human inputs such as category names or descriptive phrases while also outlining bounding boxes around the detected objects (Liu, Zeng, 2023). SAM (Segment Anything) is an

object detection model designed by Meta, formerly Facebook, that is known to produce high quality object masks from input prompts such as points or boxes (Kirillov, Mintun, 2023).

Initially, there were some hesitations on implementing GSA since the confidence score of the prediction using the GDINO model was rather low. However, when looking at the resulting segmented images, the masks drawn on the objects appears to be much more accurate and does not reflect the score produced from GDNIO. A mask in this context refers to a colored overlay on the resulting image that shows the class that is being predicted. Originally, the GSA results showed a single score, which we believed to be the GDINO score. After some further investigation, it appeared that SAM does indeed have their own confidence score independent from GDINO. Rather than using the GDINO score as the determining factor of how accurate the model is, the GSA codebase was modified to only show the prediction scored produced by SAM. This change showcased just how much more impressive these scores were compared to the GDINO scored.

3.2 3D Semantic Segmentation

The proposed dual stream pipeline includes a 3D semantic segmentation network as the second stream. KPConv (Thomas, Qi, 2019) has been selected as the backbone neural network to fulfill the role of this second stream network due to its' promising performance over several point cloud datasets. KPConv is an abbreviation for Kernel Point Convolutions, in which the program systematically breaks a point cloud into a grid, then places a "kernel" with a set radius that then takes in all the points within the radius, averages their weights out, then outputs a predicted point with a certain level of accuracy. This point will be iterated over multiple times until a consensus of what type of object this point is, is determined. This process will then be repeated over and over, across all points within a given dataset, until the total number of iterations are met, at which point, an overall accuracy is produced, as well as a predicted point cloud. This predicted point cloud can then be loaded into a software, such as CloudCompare (CloudCompare, 2024), to visualize and compare to the human-annotated ground truth. To capture the accuracy of the predictions, each point is compared to the ground truth, and determines if it is the same as the ground truth, or different. If it is different, then that point will be a different scalar value from the ground truth, allowing for easy interpretation, whereas if it is the same as the ground truth, that point will be indistinguishable from the accurate ground truth.

3.3 Ensemble Learning

Ensemble methods in machine learning are techniques that combine multiple models to improve the overall performance and robustness of a predictive model. The main idea is that by combing predictions of multiple different models, it is often able to achieve better results than a single model (Demir, 2016). This is done by focusing on mitigating the errors that reside in the individual models then deciding based on the overall scope of predictions.

3.3.1 Ensemble Learning Advantages & Disadvantages

Advantages of using ensemble methods over a singular model include an increase in accuracy in the cases where the individual models may have a high bias or variance. Another strong suit of ensemble methods is they are less susceptible to overfitting as not one model is making the prediction but a multitude of different models, where each additional model being less likely to have an overfit configuration. Lastly, ensemble methods can be used on a wide-ranging problem set that can tackle a multitude of problems at once. In our case, that is predicting on 2D and 3D simultaneously to ultimately achieve a better prediction result.

Disadvantages of ensemble methods are more computationally expensive compared to individual models (Soni, 2023). The more complex your ensemble of models, the more time is needed to process for training. Whether this be the total number of models or the architecture type of your model. The needed allotted time to fine tune hyperparameters such as batch size, epochs and learning rates scales with this complexity as well. Making changes to the combined models to fit a particular dataset is more troublesome in this case since changes to these hyperparameters are affecting the results of multiple models, thus having compounding effects on the outcome. Likewise, results from an ensemble framework usually are harder to decipher as the final prediction is a conglomerate of multiple different models.

3.3.2 Ensemble Learning Techniques

There are many ways in which to enact an ensemble learning solution. The most popular techniques, however, fall into the overarching categories of either boosting or stacking.

Boosting is an ensemble method in machine learning where multiple weak learners (models that are only slightly better than random guessing) are trained sequentially (Soni, 2023). This means that each subsequent model focuses on correcting the errors of the previous model(s) before it. The final prediction is typically made by combing the predictions of all the weak learners, often through a weighted sum. For example, a boosting technique could start with the training set which is then subdivided into an "m" number of subsets. Each separate subset is then sent to its respective weak learner to train over that section of data. After the model has learned over all subsets of data, the output is an overall prediction by a weighted average decision.

Stacking is an ensemble method in machine learning that utilizes multiple different trained models (often heterogeneous) and then combining their predictions using a Meta learner. This Meta learner is known to be a higher-level learner that uses the predictions of base models as features for training. In Figure 2, the training set is sent out, in full, to separate models that make their own separate predictions on the training set. Then afterwards these results are then used to create a new training set where the predictions are the input tensor into the final Meta model which makes a final prediction by weighted averaging.

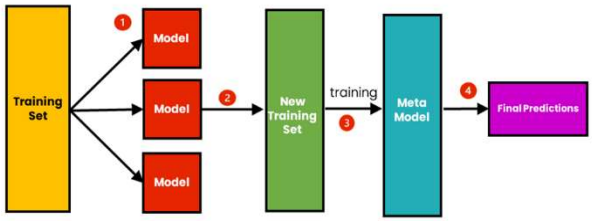

Figure 2. Stacking Pipeline

The boosting technique has some advantages over stacking. These include error reduction in standard machine learning algorithms such as linear or logistic regression. Implicit feature selection by assigning higher priority over features that are more informative in the prediction task. Also, since boosting aims to correct the errors of the models previously ran on the data, it can perform better on noisy data. With its advantages, however, some disadvantages are notable compared to stacking. Stacking is usually not as computationally expensive as boosting. Stacking also has more flexibility in model selection, especially important in the proposed solution with using 2D and 3D models. Stacking also is not prone to overfitting as it allows for a diverse selection of models which can interrupt the sequence of overfitting commonly found in the boosting technique.

Since the base classifiers are not applied sequentially, employing a boosting approach is not suggested. On the other hand, availability of training data suggests that stacking (Soni, 2023) would be the best option here. In the presented solution, we opted for a stacking ensemble method over other ones such as bagging or boosting. This gives us flexibility with the model types and the ability to use some of the strong models that we currently have, such as our 3D semantic segmentation model like KPConv. For stacking, we can have multiple strong base models with high accuracies, as well as enabling the use of a heterogeneous model structure that would otherwise be impossible under other ensemble methods.

4. EXPERIMENTAL RESULTS

In this section, we will delve into the experimental results for each individual stream (base classifiers) as well as the performance of the final ensemble classifier (Meta classifier) over three challenging datasets. Furthermore, we will demonstrate the advantage of employing stacking as opposed to solely relying on individual base classifiers, while also addressing the challenges and weaknesses that we encountered.

4.1 Datasets

To determine the efficacy of the approach, three datasets were chosen as general baselines to compare, both against one another, as well as against different experiments within each framework. The three baseline datasets are as follows: Stanford Large-Scale 3D Indoor Spaces Datasets (S3DIS) (Armeni and Sener, 2016), JBLM Hospital, and JBLM Municipal. JBLM Hospital and JBLM Municipal datasets were gathered in-house.

4.1.1 S3DIS

S3DIS is a publicly available dataset that is generally well regarded as one of the best 3D datasets, especially for indoor scenes of populated buildings by consensus. In 2017, researchers from Stanford University released the 2D-3D-S dataset to the public. Within this dataset, both 2D and 3D modalities were captured simultaneously over "3 different buildings of mainly educational and office use" (Armeni, Sax, 2017). This capture process covers 6,020 square meters of Stanford's campus, resulting in 70,496 total 2D images, and 5,978 total 3D objects of interest, seen in Table 1. Having so many corresponding 2D and 3D samples allows for both the 2D and 3D components of the pipeline to have the same reference point in 3D space. Due to this, these samples have all the points in both 2D, and 3D space, aligned. This luxury allows for the team to have a simplified "apples to apples" comparison for performance when looking at the results.

Table 1. S3DIS Dataset Breakdown

Area	S3DIS Total Area (in Sq. Meters)	Total S3DIS 2D Images	Total S3DIS 3D Samples
1	965	10,327	925
2	1,100	15,714	1,307
3	450	3,704	460
4	870	13,268	963
5	1,700	17,593	1,437
6	935	9,890	1,001
Total	6,020	70,496	5,978

The S3DIS dataset covers 13 different object classes across 11 major room types. These classes formed the basis of the classes that were used for the extent of the experimentation. As S3DIS was the baseline for performance across both dimensional modalities, when further experimentation was done, the same list of classes was referenced when annotating other datasets for semantic segmentation.

4.1.2 JBLM Hospital

The JBLM Hospital is a 3D scan from the Joint Base Lewis-McChord (JBLM) in Washington state. This military base contains a training facility for Military Operations on Urban Terrain (MOUT). These landscapes can help soldiers train in real-world environments in the safety of a military base. The JBLM Hospital, referenced as Hospital from this point forward, mimics an urban hospital in a more remote portion of the world. Due to this, the Hospital itself appears mostly abandoned, with mostly wooden features such as tables and chairs, with the occasional hospital bed for authenticity. The Hospital covers a single floor, with two entrance points, a main waiting area, a surgery center, and a general infirmary. The Hospital is organized into seven different areas, that are then further broken down based on the objects/features in each, resulting in 143 total samples. The seven areas were selected to keep as many whole features as possible, so the divisions between these rooms are natural divisions- such as walls or entrance ways. Splitting the dataset up in this way allows for the largest subset of training material for the networks, while still preserving some logical divisions for human visualization.

4.1.3 JBLM Municipal

The JBLM Municipal is from the same JBLM military base as the Hospital, except it serves a slightly different training purpose. JBLM Municipal, referred to as Municipal, resembles a three-floored apartment building, used for training the military in close-quarters residential scenarios. There are three floors that consist of stairs, empty rooms, and hallways, some of which are railed walkways overlooking a central courtyard. Municipal is broken up into its three constituent floors, which are then further broken down into rooms, then into the desired objects of interest, resulting in 475 samples. The Municipal building appears abandoned as evidence by completely empty rooms that have been worn down to the bare concrete. Additional breakdown of all datasets will be provided in Table 2.

4.1.4 Objects of Interest

During the annotation process, it is important for the list of objects of interest to be determined so that all objects can be appropriately classified. Seeing as how S3DIS is the initial dataset that was used for all baselines, the class list was already defined - 12 classes with one added as a "catch-all" class defined as "Clutter". This class list was then used when defining the objects for other datasets. As the project progressed, and requirements were tweaked, an additional class was added that was not formerly desired- doorframe. Prior to the addition of this class, S3DIS annotated doorframes strictly as part of the door. To avoid having to re-train the 3D network to be able to detect this new class separately, due to the time constraint of the project, it was decided to continue to use the pre-trained model as-is. A full breakdown of all the datasets, and the number of classes within each is shown in Table 2. It is worth noting that some objects in Table 2 are not commonly found in JBLM Hospital and JBLM Municipal, noted by the -.

4.1.5 Annotating 3D Point Cloud Datasets

For any machine learning process, having an accurate ground truth is vital. For S3DIS, this was not a concern due to the developers of the S3DIS dataset having already gone through and annotated every individual pixel to the best of their abilities. For Hospital and Municipal, this was not already done and needed to be performed by the team. The general process of annotation followed these steps: 1) Open a software that visualizes point clouds and allows for manipulation and editing of the point clouds (CloudCompare in this case). 2) Load in the large point cloud. For Municipal and Hospital, this meant loading up the entire building, and breaking the point cloud up into constituent floors and then rooms. 3) Further break down rooms into the key objects of interest seen in Table 2. If an encountered object is not in the list of objects of interest, then it is lumped in the "catch-all" class that is "Clutter". 4) Repeat until all rooms are annotated, and all objects within the rooms have been given a class. After annotation, reorganization of the files is occasionally necessary to match the structure of S3DIS. When reorganizing the files, it is in the context of the files within the file directory. If a folder does not have the correct structure, KPConv will crash as the program is unable to find the necessary files. This step could have been skipped, but it allows for an easy and seamless transition between different datasets when using KPConv.

Table 2. Number of Samples per Object Class

Classes	S3DIS	JBLM Hospital	JBLM Municipal
Ceiling	385	7	46
Floor	284	7	42
Wall	1,548	62	181
Beam	159	-	-
Column	254	-	-
Door	543	14	35
Window	168	10	57
Table	455	6	-
Chair	1,363	6	-
Sofa	55	-	-
Bookcase	583	1	-
Board	137	-	-
Clutter	3,882	13	35
Doorframe	-	18	79
Total Number	9,816	144	475

4.1.6 Annotating 2D Images

Much like the 3D Annotations, annotating the 2D images of the similar dataset is essential as it provides the ground truth for the objects that are to be detected. For the contents of the images to be the same as the 3D point cloud dataset, we took inspiration from Jiuyi Xu, Meida Chen and the team over at the Institute of Creative Technologies for their implementation of 2D projection into 3D space (Xu, Chen, 2024). The process starts with rendering a OBJ mesh file, that is then systematically iterated through at a set interval to capture 2D images. After these images are captured, an online annotating tool such as Roboflow (Dwyer, Nelson, 2024) is used to annotate these images. While Roboflow can provide useful features, such as auto labeling and smart polygons, which helps expedite the annotating process to some degree, the fact the images are rendered means that there are bound to have incidents of some of the objects being distorted. Because of the distortion of these images, there were a good amount of misclassification (i.e.: objects that were "walls" ended up detecting as either "windows" or "ceilings"). This requires the user to rectify any misclassification by manually reannotating the detected objects.

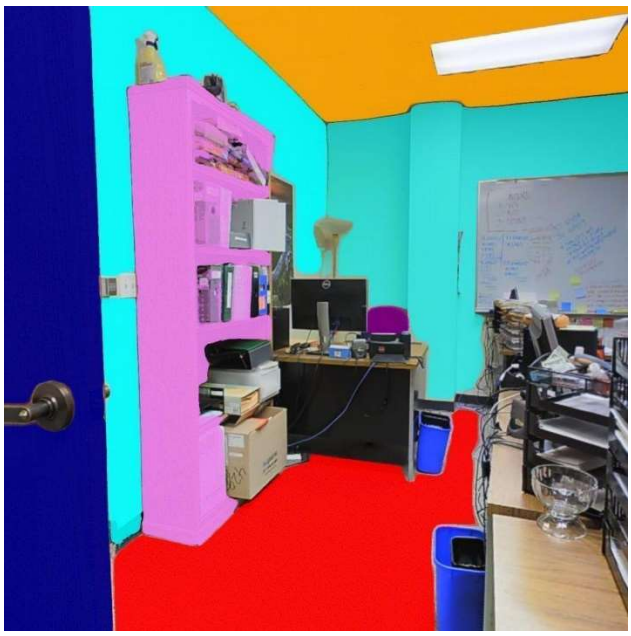

Figure 3. GSA Segmentation Results

4.1.7 Aligning the Datasets

An integral part of the data pre-processing stage is weakly aligning the datasets in 2D, and 3D space. Mentioned previously, the 2D images are captured via rendering of an OBJ mesh file. Each of these images will contain a set number of pixels (resolution of the image) and these pixels are then "projected" out into 3D space onto the underlying 3D point cloud underneath the façade of the OBJ mesh. If a pixel's projection aligns with a point in 3D space, this point is then saved out to a new

point cloud. This new 3D point cloud is referred to as weakly-aligned with the 2D image.

This weakly-aligned point cloud is then passed to the 3D semantic segmentation network, and the images are passed to the 2D semantic segmentation network. The output of these processes is still weakly-aligned. For the final ensemble portion to be completed, a ground truth for both 2D and 3D is required, meaning that they also need to be weakly aligned. To do this, a multi-step process is implemented. Initially, a copy of these original files must be created. These copies are then sorted based on their contents in ascending numerical order, then a recursive binary search is called to go through the original projection weakly aligned 3D output line by line and search these new sorted files to find a match. These matches will then become the weakly-aligned ground truth. Now, having both a weakly aligned prediction and an accompanying ground truth, these files are then passed off to the ensemble segment for final predictions.

4.2 2D Semantic Segmentation

Table 3. Weakly-aligned 2D Accuracy Across the Different Datasets

Classes	S3DIS	Hospital	Municipal
Ceiling	96.04%	92.07%	98.65%
Floor	99.52%	99.08%	94.75%
Wall	96.81%	95.12%	96.65%
Window	N/A	91.96%	N/A
Door	98.47%	82.53%	93.34%
Chair	96.56%	N/A	N/A
Bookcase	94.60%	N/A	N/A

As part of the pipeline, Grounded-Segment-Anything (GSA) is used as the first network stream (base classifier) to run the 2D semantic segmentation component of the two-stream framework. There are some modifications made to the initial implementation, such as being able to predict and draw segmentation masks on multiple images and ensure that the segmentation mask for the detected object have their own specific color. The first test was conducted on a subset of the 2D S3DIS dataset and since the pretrained models for both GDINO and SAM has been provided, there was no need to run the training procedure. To expedite testing time to ensure that the model is working properly with all modifications, only six objects from Table 2 were used for this test experiment- ceiling, floor, wall, window door, chair, and bookcase. As shown in Figure 3, the generated semantic segmentation mask for the above-mentioned object classes looks promising as there aren't any signs of misclassification. After the successful result on testing on the S3DIS dataset, sample images from the other two dataset, JBLM Hospital and JBLM Municipal, were used as well.

4.3 3D Semantic Segmentation

After deciding on KPConv as the network for the 3D segmentation, the task was then pivoted to which dataset would be best to train the network on for KPConv to be able to identify building interior objects. At this point, the datasets that KPConv was tested on in the original paper (Thomas, Qi, 2019) were analyzed individually. The criteria in which they were analyzed namely consisted of the list of classes that they contained. Some were more of a generalized object list i.e. ShapeNet (Chang, Funkhouser, 2015), whereas others like ScanNet (Dai, Chang, 2017), offer scenes of interiors with lots of objects, but not larger interior spaces. For this task, the best dataset that accompanied KPConv was S3DIS.

Now having both a network and a dataset to train over, the first step was to reproduce the results found in the KPConv paper (Thomas, Qi, 2019). KPConv reported an IoU of 67.1% when training over Areas 1-7, omitting Area 5, and then testing over Area 5 of S3DIS. It is important for valid results for the test set of any machine learning network to be completely unseen. When the team ran the same training and testing configuration, an IoU of 66.14% was achieved. This IoU is lower than the

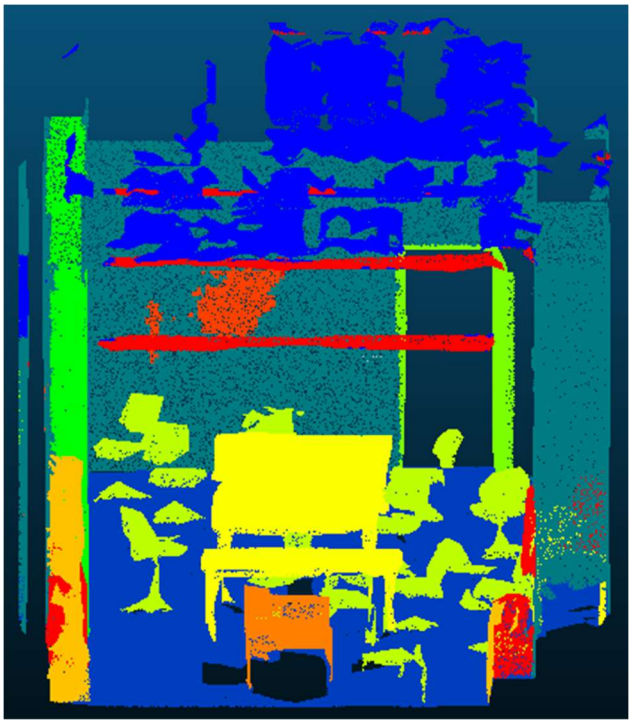

Figure 4. KPConv Segmentation Results over S3DIS

original reported IoU, but still within an acceptable margin of error to be considered run-to-run variance which is normal for machine learning tasks. Additional tests were performed, changing parameters for training, but the default configuration of 500 epochs, 500 epoch steps, with a batch size of 6, provided the best results. This configuration allowed for KPConv to run on a variety of machines, without running out of memory on the GPUs. As the batch size increases, more memory is required per-epoch. Due to the "consumer" nature of the computers used for this experimentation, GPU memory is a scarce commodity, meaning that it was found to be better to run a smaller batch, with an increased number of epochs to attempt to compensate for this limiting factor.

Once we were able to reproduce the results reported from KPConv originally, moving to test other datasets was the next step. The Hospital dataset was the next one to work with. Initially, the Hospital was only tested on using the pre-trained S3DIS model mentioned previously. This resulted in an extremely positive score of 76.89% IoU. Seeing this score gave us a very positive outlook on the future of the project. This positivity was short-lived, however. The next step was to attempt to see if training KPConv from scratch and then testing over the Hospital would yield just as good of results as S3DIS did. When this was trained, the model's accuracy dropped to 49.13% IoU. This potentially could have been due to over-training of the model on such a small dataset, but the S3DIS model seemed to provide a good balance in terms of objects of interest, as well as having so many samples to train with.

Table 4. Weakly-aligned 3D IoU Scores Across the Different Datasets

Classes	S3DIS	Hospital	Municipal
Ceiling	90.30%	95.18%	50.84%
Floor	95.25%	94.86%	21.57%
Wall	72.55%	88.27%	83.38%
Beam	20.32%	N/A	3.09%
Column	0.16%	N/A	42.06%
Window	22.01%	0.0%	0.0%
Door	47.67%	46.66%	22.26%
Chair	95.22%	74.46%	9.85%
Table	53.24%	48.29%	N/A
Bookcase	25.96%	18.43%	0.0%
Sofa	9.55%	N/A	N/A
Board	2.54%	N/A	N/A
Clutter	39.20%	27.88%	15.12%

Following the tests with the Hospital, a similar procedure was performed with Municipal. Step one was performing transfer learning to test the pre-trained S3DIS model resulting in a score of 70.04% IoU. This further emphasized the validity of the model and its robustness when testing over new, never-before-seen datasets. The next step was to then attempt to train the model using Municipal itself. Initially, this was a bit difficult. Originally, we used floors to determine what was used for training, and what was used for testing. This meant that out of the three-floored Municipal building, only one floor would be used for testing. This resulted in an extremely poor 45.62% IoU. Seeing this poor score, a custom split was devised to maximize the number of training samples compared to the testing samples**.** This custom split re-structured the dataset directory to test on 3 rooms, and train on 39. These testing samples were specifically picked to maximize the number of objects of interest per sample. Using this new testing set, an IoU score of 70.04% was achieved. This IoU is still not as good as the pre-trained S3DIS model, meaning that S3DIS is the best choice for the initial training of our model, due to its substantial size and diverse set of classes.

Now that an ideal model was selected, that model had to then test over the unseen weakly-aligned samples. These samples had less points (due to the projected nature of the point clouds), so some areas were significantly sparser than others. Due to this sparse nature, KPConv had a much harder time attempting to identify objects. As a result, the 3D portion of the experiment performed much more poorly compared to the earlier mentioned baselines. Table 4 showcases the per-class IoU scores of the datasets, and there are some classes, such as "Window", that the 3D model could not identify properly at all. These weakly-aligned predictions that were an output of KPConv, were then passed into the ensemble portion. These files still contain the same points as they did prior to 3D predictions. It was also ensured that each point in both the prediction and the 3D ground truth were in the same order and contained the same points that they originally did when the 2D images were captured.

4.4 Ensemble

The intuition for the choice of the two base classifiers is that their diverse nature makes them very much complementary, and hence extremely suitable for combining in an ensemble learning method to achieve a performance superior to both methods. With the base models configured and fine-tuned, we set out to find the most optimal ensemble learning implementation. Since the problem set calls for a classification of objects, we opted for a logistic regression model which will intake the predictions made by the 2D (GSA) and 3D (KPConv) semantic segmentation

solutions and produce a classification prediction itself. Each set of predictions made separately by the 2D, and 3D models will be represented as an array. These two arrays will be aligned along with the ground truth representing the newer Meta Dataset. Following this newly formed dataset, it will be sent into the Meta model as input giving the final predictions.

After the merging of S3DIS dataset, the resulting number of points equaled to 972,810 samples with all overlapping ground truths. The resulting predicted classes from the 2D solution is (0, 2, 5, 6, 7, 8, 9), whereas KPConv solution contained all found classes in the ground truths (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12). It should be noted that class numbering here is 0-based, meaning that ceiling, the first class identified in the class list in Table 2, is represented as 0, and each class iterates by one up from there. The final accuracy for the Meta Classifier over S3DIS is 94.63%. This is a definite improvement in comparison to the 3D solution which had a resulting accuracy of 93.97%. Using the ensemble methodology gave a slight increase in accuracy over the 3D solution for S3DIS.

The Meta Dataset of the Hospital dataset ended up with 5,313,199 points with the intersection of ground truths being (0, 1, 2, 5, 6, 7, 8, 9, 10, 12). The unique prediction classes for the 2D solution came out to be (0, 2, 5 6, 9) and the 3D solution having (0, 1, 2, 5, 6, 7, 8, 9, 10, 12). The final accuracy for the Meta Classifier is 98.05% on the Meta Dataset. This is an improvement to both the 2D and 3D solutions on their own, more so in the case of 3D where we have an increase in accuracy of 2.56%.

The merging of the Municipal Meta Dataset left 10,881,524 data points found and common ground truths being (0, 1, 2, 3, 4, 5, 6, 7, 9, 12). The unique prediction classes for 2D were (0, 2, 5, 6, 9) while the 3D was (0, 1, 2, 3, 4, 5, 6, 7, 9, 12). The final accuracy over Municipal for the Meta Classifier was 93.59%. This result was an improvement to the 3D solution by 3.09%. Table 5 summarizes the performance of the base 2D and 3D models as well as the stacking Meta classifier over classes that make over 10% of the testing dataset. This focus on the classes with a 10% minimum helps filter out classes that were extremely sparse because of the projection method.

A trend with the ensemble solution implementation is that compared to the 3D solution there is always an increase in accuracy. With S3DIS there is a slight increase in accuracy, whereas Hospital and Municipal have a more substantial increase in accuracy. Even with the 2D solution not containing all unique prediction values to ground truths within each Meta Dataset, there is still a favorable result in comparison to KPConv alone. Ideally, both models would have the highest number of common objects found possible. Doing so would give a preferable result as the common points between the 2D, and 3D predictions would also have performed well on both. In the case of Hospital, there was an increase in accuracy after Stacking compared to the 2D and 3D counterparts individually. Overall, using a Meta Classifier proves to garner impressive results by utilizing both 2D and 3D networks for semantic segmentation with the Meta Classifier leveraging the strength of different modalities and enhance segmentation performance. These results, although on a small subset of datasets, show promise in the ability of stacking as a supplementary addition to 3D semantic segmentation networks to improve results, regardless of the dataset.

Table 5. Weakly-aligned Average Accuracy Across the Different Datasets

Dataset	2D (%)	3D (%)	Stacking (%)
S3DIS	94.97	93.97	94.63
Hospital	95.79	95.49	98.05
Municipal	94.93	90.50	93.59

5. CONCLUSION

Ensemble learning has gained popularity in the field of machine learning due to its ability to combine multiple models to improve predictive performance. In this paper, we propose an ensemble method to learn a Meta Classifier comprising of 2D and 3D semantic segmentation networks, the two heterogeneous base classifiers. The integration of 2D semantic segmentation with 3D networks represents a cutting-edge approach that combines the strengths of 2D image analysis and 3D data processing to enhance segmentation performance and enable more comprehensive scene understanding. Extensive experiments over three datasets demonstrated that due to their heterogeneity, the two base classifiers are complementary in terms of their per-class accuracy. Hence, the ensembles of classifiers outperform the individual 2D and 3D base classifiers. Moreover, by leveraging multi-modal learning, feature fusion, and knowledge

transfer, this hybrid approach offers a powerful tool for advancing research and applications in computer vision, point cloud analysis, and 3D scene understanding.

6. REFERENCES

Aoki, Y., Goforth, H., Srivatsan, R., Lucey, S. PointNetLK: Robust & efficient point cloud registration using PointNet (2019). Retrieved from https://arxiv.org/abs/1903.05711

Armeni, I., Sax, A., Zamir, A. R., & Savarese, S. (2017). Joint 2D-3D-Semantic Data for Indoor Scene Understanding. ArXiv E-Prints. Retrieved from http://arxiv.org/abs/1702.01105

Armeni, I., Sener, O., Zamir, A. R., Jiang, H., Brilakis, I., Fischer, M., & Savarese, S. (2016). 3D Semantic Parsing of Large-Scale Indoor Spaces. Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition.

Boulch, A., Guerry, J. Saux, B., Audebert, N. "Snapnet: 3d point cloud semantic labeling with 2d deep segmentation networks", Computers Graphics, vol. 71, pp. 189-198, 2018

Chang, A. X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., … Yu, F. (2015). ShapeNet: An Information-Rich 3D Model Repository. arXiv [Cs.GR]. Retrieved from http://arxiv.org/abs/1512.03012

Chen, L., Chen, W., Xu, Z., Huang, H., Wang, S., Zhu, Q., Li, H. Dapnet: A double self-attention convolutional network for point cloud semantic labeling, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 14 (2021) 9680– 9691.

CloudCompare (version 2.12.4) [GPL software]. (2024). Retrieved from http://www.cloudcompare.org/

Dai, A., Chang, A. X., Savva, M., Halber, M., Funkhouser, T., & Nießner, M. (2017). ScanNet: Richly-annotated 3D Reconstructions of Indoor Scenes. Proc. Computer Vision and Pattern Recognition (CVPR), IEEE.

Demir, N. (2016, February 4). Ensemble methods: Elegant techniques to produce improved machine learning results: Toptal®. Toptal Engineering Blog. https://www.toptal.com/machine-learning/ensemble-methods-machine-learning#:~:text=Ensemble%20methods%20are%20techniques%20that,than%20a%20single%20model%20would

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. 2009 IEEE Conference on Computer Vision and Pattern Recognition, 248–255. doi:10.1109/CVPR.2009.5206848

Dwyer, B., Nelson, J., Hansen, T., et. al. (2024). Roboflow (Version 1.0) [Software]. Available from https://roboflow.com

Fei, J., Peng, K., Heidenreich, P., Bieder, F., Stiller, C. Pillarsegnet: Pillar-based semantic grid map estimation using sparse lidar data, CoRR abs/2105.04169 (2021). arXiv:2105.04169. URL https://arxiv.org/abs/2105.04169

Hu, J., Kuai, T., Waslander, S. Point density-aware voxels for lidar 3d object detection (2022). doi:10.48550/ARXIV. 2203.05662. URL https://arxiv.org/abs/2203.05662

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., … Girshick, R. (2023). Segment Anything. arXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/2304.02643

Li, X., Wang, L., Wang, M., Wen, C., Fang, Y. Dance-net: Density-aware convolution networks with context encoding for airborne lidar point cloud classification, ISPRS Journal of Photogrammetry and Remote Sensing 166 (2020)

Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., … Zhang, L. (2023). Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection. arXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/2303.05499

Lu, T., Wang, L., Wu, G. Cga-net: Category guided aggregation for point cloud semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 11693–11702.

Maturana, D., Scherer, S. Voxnet: A 3d convolutional neural network for real-time object recognition, in 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)

Qi, C., Su, H., Mo, K., Guibas, L. Pointnet: Deep learning on point sets for 3d classification and segmentation, CoRR abs/1612.00593 (2016). arXiv:1612.00593. URL http://arxiv.org/abs/1612.00593

Qi, C., Yi, L., Su, H., Guibas, L. Pointnet++: Deep hierarchical feature learning on point sets in a metric space (2017). arXiv:1706.02413. 1, 4, 5, 11

Ren, T., Liu, S., Zeng, A., Lin, J., Li, K., Cao, H., … Zhang, L. (2024). Grounded SAM: Assembling Open-World Models for Diverse Visual Tasks. arXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/2401.14159

Shapiro, L., Stockman, G. (2001): "Computer Vision", pp 279–325, New Jersey, Prentice-Hall, ISBN 0-13-030796-3

Soni, B. (2023, May 1). Stacking to improve model performance: A Comprehensive Guide on Ensemble Learning in Python. Medium. https://medium.com/@brijesh_soni/stacking-to-improve-model-performance-a-comprehensive-guide-on-ensemble-learning-in-python-9ed53c93ce28

Soni, B. (2023, May 1). Understanding boosting in Machine Learning: A comprehensive guide. Medium. https://medium.com/@brijesh_soni/understanding-boosting-in-machine-learning-a-comprehensive-guide-bdeaa1167a6

Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E. Multi-view convolutional neural networks for 3d shape recognition, in 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 945–953. doi:10.1109/ICCV.2015.114. 4 [16]

Tchapmi, L., Choy, C., Armeni, I., Gwak J., Savarese, S. "Segcloud: Semantic segmentation of 3d point clouds", 2017 International Conference on 3D Vision (3DV), pp. 537-547, 2017

Thomas, H., Qi, C. R., Deschaud, J.-E., Marcotegui, B., Goulette, F., & Guibas, L. J. (2019). KPConv: Flexible and Deformable Convolution for Point Clouds. Proceedings of the IEEE International Conference on Computer Vision.

Wu, Z., Song, S., Khosla, A., Tang, X., Xiao, J. 3d shapenets for 2.5d object recognition and next-best-view prediction, CoRR abs/1406.5670 (2014). arXiv:1406.5670. URL http://arxiv.org/abs/1406.5670

Wu, Z., Yang, P., Wang, Y. Mvpn: Multi-view prototype network for 3d shape recognition, IEEE Access 7 (2019) 130363– 130372. doi:10.1109/ACCESS.2019.2937489.

Xu, J., Chen, M., Feng, A., Shi, Y., Yu, Z. (2024). "Open-Vocabulary High-Resolution 3D (OVHR3D) Data Segmentation and Annotation Framework" Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC)

Zhang, C., Luo, W., Urtasun, R. Efficient convolutions for real-time semantic segmentation of 3d point clouds, 2018 International Conference on 3D Vision (3DV) (2018) 399–408. 4

Zhou, X., Girdhar, R., Joulin, A., Krähenbühl, P., Misra, I. (2022). Detecting Twenty-thousand Classes using Image-level Supervision. ECCV. [Original source: https://studycrumb.com/alphabetizer]