Reducing domain shift in synthetic data augmentation for semantic segmentation of 3D point clouds

Romain Cazorla, Line Poinel, Panagiotis Papadakis, Cédric Buche

Reducing domain shift in synthetic data augmentation for semantic segmentation of 3D point clouds

Romain Cazorla
Segula Technologies
Lab-STICC, UMR 6285, team RAMBO
France
romain.cazorla@segula.fr

Line Poinel
Segula Technologies
France

Panagiotis Papadakis
IMT Atlantique
Lab-STICC, UMR 6285, team RAMBO
F-29238 Brest, France

Cédric Buche
CROSSING
IRL CNRS
Australia

***Abstract*—The use of deep learning in semantic segmentation of point clouds enables a drastic improvement of segmentation precision. However, available datasets are restrained to a few applications with limited applicability to other fields. Using synthetic and real data can alleviate the burden of creating a dedicated dataset at the cost of domain-shift that is mostly addressed during training, while treating the problem directly on the data has been less explored. Towards this goal, two methods to alleviate domain shift are proposed, firstly by enhanced generation and sampling of synthetic data and secondly by leveraging color information of unlabeled point clouds to color synthetic, uncoloured data. Obtained results confirm their usefulness in improving semantic segmentation result (+3.43 into mIoU for a network trained on S3DIS zone 1). More importantly, the devised coloring method shows the ability of a point-based network to link color information with recurrent geometric features. Finally, the presented methods are able to bridge the domain-shift gap even in cases where inclusion of raw synthetic data during training impedes learning.**

I. Introduction

Nowadays, there are two central domains of application for semantic segmentation of point clouds, namely, robotic vehicles and autonomous cars [30] [10]. This results in two sources of point clouds that are mostly available: those acquired from urban exterior environments or those from building interiors such as offices or living spaces [10]. In contrast, there is a lack of datasets in other applications and study cases of strong interest, such as in industrial domains where available data are proprietary e.g. in naval, energy or petrochemical industries. Admittedly though, state-of-the-art methods on semantic segmentation of point clouds rely on deep learning techniques [30] whose effectiveness is based on the use of humongous amount of data.

Creating a new dataset is a tedious task as data collection and annotation require considerable time, effort and expertise. When low resolution sensors (e.g. RGB-D cameras) are insufficient, specialized equipment (for lasergrammetry) or software and computational power (for photogrammetry) are necessary whereas annotation can take an order of magnitude longer in man hours than acquisition.

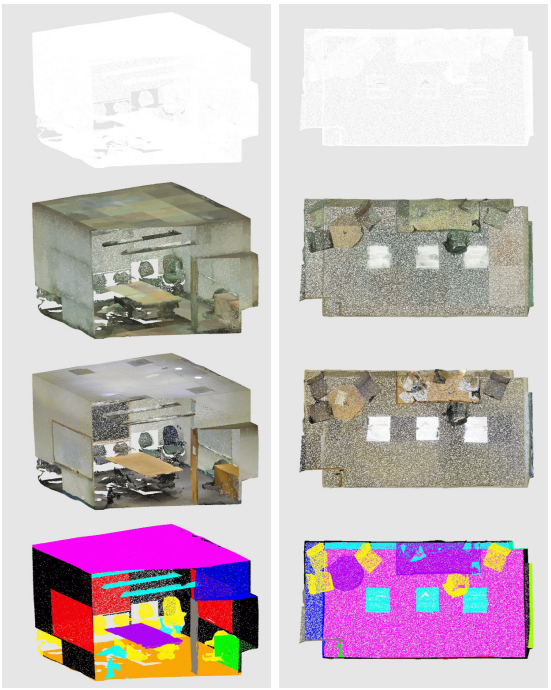

Fig. 1. Illustration of the used coloring method. From top to bottom : colorless scene, colored scene obtained by a network trained on unlabeled data, colored ground-truth scene and semantic segmentation ground truth. Coloring helps to distinguish objects of certain shapes such as lights or tables.

To limit the quantity of data to be acquired and processed, it is possible to resort to techniques such as transfer learning, [13], few-shot learning [32], self-supervised learning [31], [22], contrastive learning [14] or synthetic data augmentation [29] which is the focus of the current work. While promising it is an understudied option in the case of semantic point cloud segmentation which could be used jointly with other techniques. Also, as synthetic data augmentation precedes learning, it can have broader benefits irrespective of the subsequent training approach.

A common issue in successfully using synthetic data is the domain shift problem [21] as deep convolution neural networks do not automatically generalize on real-world data

TABLE I
COMPARISON OF DIFFERENT SYNTHETIC SEMANTIC SEGMENTATION DATA METHOD CREATION.

Paper	Data source	Virtual acquisition method	Color
SqueezeSeg [28]	Video Game	Ray tracing + post processing	From source
SqueezeSeg V2 [29]	Video Game	Ray tracing + post processing + learned intensity	From source
SceneNet RGB-D [16]	Randomly created scenes	2D camera	Texture library
BIM-To-Scan [18]	CAD models	Full wave simulation [27]	None

when trained on synthetically generated ones. Most solutions in 2D focus on the use of methods to align the distribution of computed features in both spaces [21]. Nevertheless, before applying such methods, it should be possible to leverage the nature of point cloud data by directly working on the creation process to reduce the domain distance. This creation process can be divided in three phases, each bearing different opportunities to improve synthetic data. The first concerns the scene creation process, where the layout and objects used are determined. The second phase concerns the point cloud generation process from the created scene that can range from simple sampling used by classification datasets to simulated LiDAR acquisition. Last but not least is the addition of new features to the point cloud such as colors (Fig. 1), normal vectors of surface orientation or intensity.

Following up to our earlier findings [7] related to major bottlenecks of semantic segmentation of point-clouds representing industrial scenes, we propose in this work a three-step creation pipeline with contributions pertaining to the last two phases, namely[1]:

- The use of synthetic 3D models for data augmentation is promoted and a quick scene creation protocol is proposed.
- The influence of 3D point sampling schemes is studied.
- A novel coloring method is presented which allows the exploitation of colored, unlabelled, data.

II. RELATED WORK

Currently, datasets for semantic segmentation of point clouds are restrained to only a small number of application domains (Fig. 2). Deep learning methods operating on point clouds can be divided in two categories depending on the way that they operate on the point cloud data [10]. Projection methods transform a point cloud into another format where more conventional methods can be used (such as 2D images [3] or voxels [17]). On the other hand, point-based methods work directly on the cloud, either by using Multi-Layer Perceptrons [19], custom convolution kernels [25] or a graph of the cloud [26]. In the case of multi-view methods, transfer learning can be used on the 2D segmentation network thanks to the large quantity of available datasets. Alternative data sources need to be sought to compensate the lack of training data for the other methods.

In the sequel, a brief review of the current use of synthetic data augmentation methods for point cloud semantic segmentation is provided (Tab. I), followed by a review of approaches for coloring a point cloud.

[1]Codeavailableathttps://github.com/RomainCazorla/Synthetic3DPointCloudDomainShiftReduction

A. Synthetic data augmentation

When considering works on point cloud at large, two of the most used datasets are in fact based on meshes transformed to point cloud with a random surface sampling : ModelNet [33] and Shapenet [8]. However, they are both used for classification problems and it is shown that methods working on the domain shift problem for 2D images classification do not work for image semantic segmentation [21].

Previous research in devising approaches for creating realistic synthetic data for 3D semantic segmentation can be found in (Tab. I). However, those works are mostly centered on cases where realistic 3D scenes are available such as video games [28] [29]. Models from a Computer Assisted Drawing software (CAD) can also be used [18] but CAD meshes are sometimes symbolic instead of realistic and the represented scenes are mostly proprietary. Finally, creating scenes in a stochastic manner is possible and was used for an RGB-D application [16]. Transferability of this method to point cloud has yet to be studied.

In the case of SqueezeSeg [28], realistic sampling is not the core of the work and a simple ray tracing process is used. However, noise is simulated upon acquisition and the second version of SqueezeSeg [29] introduces methods to reduce domain shift. First, focal loss is used to alleviate the data imbalance created by the synthetic data which would introduce too much background information otherwise. A learning process to add intensity to the point cloud is also used. In both versions, color information is directly drawn from the scene model.

On the other hand, the study realised by Noichl et al. [18] focuses on the sampling process. The case study shows the success of full-waveform simulation [27] in reducing synthetic to real data distance. However, the influence on semantic segmentation is not studied and the color information is not considered in their work.

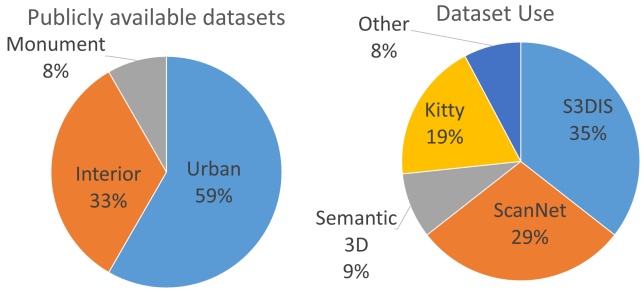

Fig. 2. Distribution of semantic point cloud segmentation datasets and their relative use in top-tier articles spanning a 4-year period since 2017. Indicative statistics for area of interest.

B. Point cloud coloring

The ability of the color channel to convey beneficial information for semantic segmentation of point clouds when using deep learning techniques, is seldom studied in the current literature. Few works can be found on RGB-D data coloring such as De²CO [6] that propose a meaningful use of coloring to convey depth of an image.

To the best of our knowledge, a coloring method applicable to an entire point cloud scene without prior knowledge of the color channel does not exist in the current literature. PCCN [15] and [4] use a Generative Adversarial Networks (GAN) to colorize the point cloud but are limited to single objects. More recently, the Point2color [23] method, also based on a GAN, allows for the coloring of Airborne LiDAR scenes. However, this method uses two discriminators, one working directly on the cloud and the other on a projected image of the airborne view of the scene. Finally, style transfer methods could be relevant to the task at hand. PSNet [5] proposes a network able to transfer the color style of an image or point cloud to an entire point cloud scene. Regretfully, the loss used in PSNet necessitates prior knowledge of the content, including colour, of the modified scene.

III. Approach

A synthetic data generation method composed of three steps is introduced : generating a scene composed of 3D meshes (Sec. III-A), transforming the scene in a point cloud (Sec. III-B) and then coloring the obtained scene (Sec. III-C).

A. Synthetic data generation

To generate synthetic 3D scenes, a method inspired by [16] and [18] is used. Each scene is created by first choosing a random layout. These layouts are 3D models defining the scene structure: walls, floor, ceiling, doors, windows and sometimes miscellaneous objects such as lights or decorations on the wall. The space in the center of these layouts is empty to allow content addition. Once the layout and its bounds are known, the scene is populated with objects by using a physics-engine [24].

In [16], a certain density of objects was desired with bigger objects less frequently used than smaller ones. However, due to the stochastic nature of the positioning process, a high number of these objects were discarded at the end of the simulation due to their fusion. To simplify and accelerate this process, scene generation is split into four phases, starting with layout selection where only the boundaries are kept followed by large objects positioning. Once these are settled, the smaller ones are added within the layout boundary, adding the complete layout geometry, which in the end can contain additional clutter objects. This sequential process increases the probability of object stacking, leading to a more complex scene (Fig. 3).

Additional controls are subsequently employed for early removal of fused objects. Increasing friction forces allows for important reduction in bouncing computation at the expense of making objects sometimes positioned at an unnatural angle (i.e. with only one of its feet on the ground). Upon the end of physical simulation, the position and the orientation of each object are automatically checked to correct defects such as unnatural angles due to increased frictions.

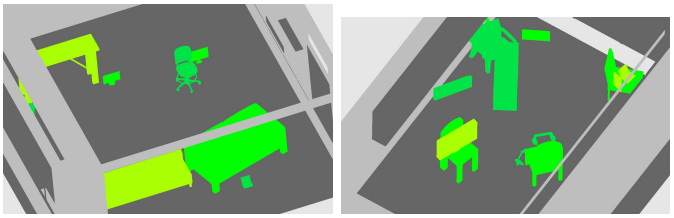

Fig. 3. Examples of generated scene. Without (left) and with (right) the sequential process.

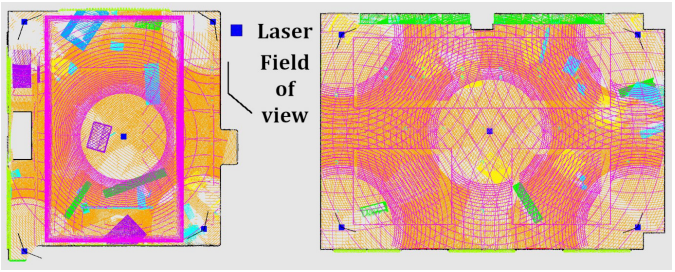

Fig. 4. Generated scenes with laser positions used for scanning, an absence of field of view markers indicate a 360° laser scanning.

While this yields a physically plausible scene it is not necessarily a human like configuration of objects (see Fig. 3 and Fig. 4 for examples of generated scenes). In contrast to [16], the sequential object placement process allows to place objects of some classes on top of others, for a more complex and realistic setting.

B. Sampling

The next step consists in obtaining a point cloud representation out of the 3D scene mesh. In contrast to [28] [29], the used sampling process closely resembles an acquisition via lasergrammetry [27].

First, a terrestrial rotating laser is defined with characteristics mimicking conventional equipment such as the minimum range, beam divergence and ranging accuracy (Tab. II).

Depending on the layout, different laser positioning strategies are used. For layouts comprised of only one, rectangular room, the laser is placed at its center and at each corner. When positioned at a corner, a laser has a 130° horizontal field of view oriented towards the room center. When positioned at the room center, a 360° horizontal field of view is used by the laser (Fig. 4). In case of more complicated shapes, the same rules are applied except when a small wall (<3 meters)

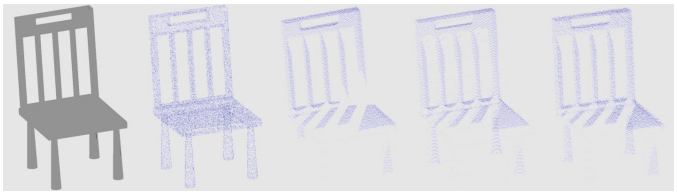

Fig. 5. Comparison of different sampling methods, from left to right : original mesh, random surface sampling, long range laser, medium range laser, short range laser. Virtual laser configurations are detailed in Table II.

TABLE II
SPECIFICATIONS OF THE VIRTUAL LASERS USED FOR SAMPLING

Laser	min range (m)	accuracy (m)	beam divergence (rad)
Long range	1.5	0.005	0.0003
Medium range	0.6	0.003	0.0042
Short range	0.13	0.01	0.001

is present in the scene, in which case the laser is positioned at its center with a $180°$ horizontal field of view.

With layouts composed of several rooms, two positions are used for each room. The two points are on the first and second third of a line positioned at the center of the room along its longest side. Each laser has a $360°$ horizontal field of view. When a room occupies more than half of a layout, it follows the rules described in the previous paragraph.

This limitation on horizontal field of view increases the effectiveness/computing cost ratio of sampling. It also helps in obtaining a relatively dense point cloud which contains scanning artifacts such as occlusions, noisy surfaces and abrupt point density variations compared to the more direct method of random surface sampling used for classification problems [19] (see Fig. 5 for a comparison of different sampling schemes). Nonetheless, a non-trivial downside of this process is the time required to create the point cloud compared to a baseline random surface sampling.

C. Coloring

As one goal of this work is to enrich a colorless point cloud with color information, employing style-transfer techniques such as PSNet [5] is not feasible: the a-priori knowledge of color is absent from the targeted point cloud. A more general method is thus proposed, able to take point positions and optionally normals as inputs and output associated colors.

To perform this task, an encoder-decoder network based on the KPConv kernel (cf eq. (1) of [25]) is used. Except for the first layer, each subsequent layer of the encoder network is organised as a ResNet (figure 2 of [12]) block, with an inner skip link and leaky ReLU around a non-deformable KPConv kernel (cf eq. (4) of [25]). Skip links are used to propagate points between the encoder and decoder.

The propagation of features in the decoder is made with a KNN-interpolation followed by an Multi-Layer Perceptron (MLP) as defined in PointNet++ [20]. Finally, a last layer composed of an MLP and a sigmoid is used. The network architecture is summarised in Fig. 6.

Treating the problem of color generation as a regression problem, a mean-squared error (MSE) is used as the loss function. If P points are considered, with c and cp their real color and predicted color vector respectively, the loss is expressed as :

$$loss = \frac{\sum_{i=1}^{P} ||cp_i - c_i||_2}{P} \tag{1}$$

This loss presents two advantages. First, it can be used for arbitrary color format as long as consistency is maintained in the data. Secondly, it ensures a reasonable correlation between

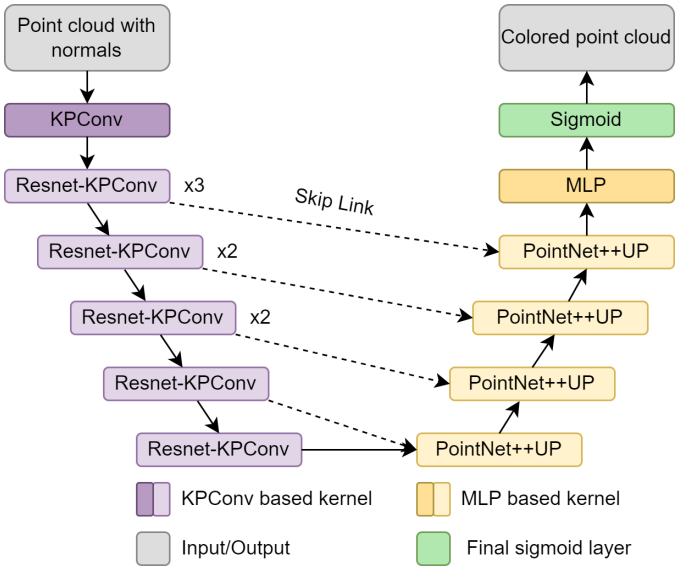

Fig. 6. Illustration of the coloring network used.

positions and colors. Specific and recurrent spatial configurations will receive distinctive coloring as MSE strongly penalizes large discrepancies. Whereas indistinct spatial configurations will get a neutral coloring which is favored by MSE that mitigate small errors. In parallel, this tolerance could also lead to small perturbation in the color channel thus reducing the risk of over-fitting.

However, it does not guarantee a full use of the color spectrum. At first, this seems to be a clear drawback, two regularization strategies are thus explored. The first, called hereafter distance regularization, is penalising the lack of use of the brightest and darkest colors; it concentrates the color spectrum around the center with a marginal use of the rest of the spectrum. When considering S as the wanted color vector spectrum and $\widehat{S}$ as the one obtained:

$$\begin{aligned} R_{dist} = &||min(S) - min(\widehat{S})||_1 + \\ &||max(S) - max(\widehat{S})||_1 \end{aligned} \tag{2}$$

The second one, called below variance regularization, adds the difference between the predicted color variance and the true color variance, which constrains the network to use more of the color spectrum, with sometimes a less meaningful coloring.

$$R_{var} = ||\sigma^2(S) - \sigma^2(\widehat{S})||_1 \tag{3}$$

IV. EXPERIMENTAL RESULTS

A hypothetical scenario where a small quantity of labeled data and a larger quantity of unlabeled data are available is used to evaluate performance. Experiments are performed in the S3DIS [2] dataset which is divided in 6 zones representing an office-like space and contains 273M points. To simulate a lack of labeled training data, only the first zone is used for training the segmentation network. This zone covers $1442m^2$ and contains 44M points. As is usually done, the fifth zone is used for testing ($2520m^2$ and 79M points) while the remaining

TABLE III
DIFFERENCES OBTAINED BY CHANGING THE VIRTUAL LASER USED IN THE SAMPLING METHOD.

Virtual laser	OA	mIoU	mIoU*	mVar	mVar*	beam	board	book.	ceiling	chair	clut.	door	floor	table	wall	column	sofa	window
Long range	72.2	33,6	42.5	**0.9**	**1.2**	**0,2**	18.3	42,3	69,6	52,5	34,3	35,8	77,2	55,4	66,2	3,8	**11,2**	4,3
Medium range	72.5	33.6	42.5	1.1	1.3	0,1	16,1	41,5	70,6	54,6	35,1	**36,9**	77,4	55,4	65,6	3,8	9,3	4,4
Short range	**72.6**	**33.8**	42.5	1.2	1.5	0,1	14,1	42,1	**71,2**	56,4	35,1	35,2	**78,0**	**56,8**	65,4	4,7	8,7	4,8
Zone 1 Only	67,6	31,7	39.9	2.3	2.9	0,1	**18,7**	**44,2**	60,7	**61,5**	**37,1**	27,7	46,4	53,6	**68,0**	**5,0**	7,7	**13,7**

zones are used for training the coloring network (4997m² and 150M points).

The details of the experimentation setup are first described in Sec. IV-A before generation efficiency is presented in Sec. IV-B. The influence of virtual laser configuration and coloring on semantic segmentation is studied in Sec. IV-C and Sec. IV-D respectively. Finally, an ablation study is carried out in Sec. IV-E.

A. Experimentation setup

For scene generation, the office layouts from SceneNet [11] are employed. Contrarily to the other proposed layout kinds, these layouts are mostly empty and do not contain room specific objects such as sinks. As the work is carried on the S3DIS dataset where the majority of rooms are of office type (only 5% of the rooms are neither hallways nor office-like), the remaining available layouts are not considered. Objects are taken from the ShapeNet dataset [8], the beam and column classes are not present in the synthetic data. Object size and class frequency were estimated in the zone 1 of S3DIS. During the scene generation process, class frequency intervenes first to deduce the class of each object populating the scene (whose mesh is then drawn uniformly from the ones available). Then, the estimated average object size and variance are used for normally taking samples of the corresponding random variable. Uniform distributions were used for the other random variables (i.e. choice of layout, number of objects, object position and orientation).

Three different virtual lasers are used for sampling, as detailed in Tab. II, roughly mimicking real world sensors. Namely, the Riegl VZ-400, a high-end long range laser, the Faro M 70, a lower-end medium range laser and the Matterport Pro 2, a RGB-D camera represented as a short-range laser. As the S3DIS dataset was acquired with the Matterport Pro 2 [1] which outputs meshes (that are later densely and randomly sampled to create the point cloud), the short-range virtual laser is only a best possible approximation of this equipment.

To avoid aggravating the imbalance in the S3DIS dataset, the generated scenes are divided in $1m^2$ columns. Those containing only floor, wall and ceiling are removed. This allows removing around 30% of points in most cases which would slow down training and risk degrading performance. This step is referred as reduction in the following.

Two networks are used in this experiment. The coloring network is the one described in Section III-C and the segmentation network is the KP-FCNN described in KPConv [25] implemented using the Torch-Points3d framework [9] without deformable layers. The coloring network is trained for 40 epochs as preliminary results showed no significant gain by further training. The segmentation network is trained for 300 epochs.

Results will be expressed as Overall Accuracy (OA) and Intersection over Union (IoU). The IoU is used for each class and the mean IoU (mIoU) is used for the whole testing set. MIoU* is similar to mIoU but excludes the beam and column classes as they could not be included in the synthetic data.

B. Efficiency of the proposed generation pipeline

First, synthetic data is generated for each scanner configuration. The whole generation process, on a computer with an Intel(R) Core(TM) i7-6700HQ @ 2.60GHz CPU and a Nvidia RTX Quatro 5000, generates between 5000 and 42000 points/minute after reduction. This time depends mostly on the generated scene complexity rather than the scanner used. Three quarters of this time is spent equally between sampling and coloring.

C. Influence of virtual laser configuration on semantic segmentation

Our next objective is to assess if and how a Virtual Laser Sampling (VLS) can help in reducing the domain shift problem. 3060m² of synthetic data are added to S3DIS [2] zone 1. The synthetic data are colored and only the virtual laser used change. For each case, the training is done three times and the mean result is reported Table III. Variance for each class IoU is computed and the mean variance is also reported, mVar* follow the same restriction as mIoU*.

From this study, the virtual laser settings seem to have a negligible influence on segmentation performance. The only difference being stability of result between training, with a slight advantage for the long-range scanner.

D. Influence of color and coloring on semantic segmentation

To better understand the role of color on point cloud semantic segmentation, only real data is used in this experiment. First, a case when color information is harmful is exposed (Tab. IV). With only the first zone of S3DIS as data, segmentation performance is worse when colors are used during training ("Raw"). Each of S3DIS zones is quite different and when another zone is added to the data or a smaller but more diverse dataset is used, this effect disappears. The more likely cause of this phenomenon is an over-reliance on color information when training data is poor. When color information is dropped during the test, a severe decrease in performance is observed for all training configurations. This confirms the strong reliance of segmentation networks on

TABLE IV
DIFFERENT DATA CONFIGURATIONS EXPOSING AN OVER-RELIANCE ON THE COLOR INFORMATION WHEN THE DATA USED IS LACKING; IN MIOU.

Training data \ Color info. used	Train / Test	White / White	Raw / Raw	Raw / White
Zone 1		**33.51**	**31.73**	**9.97**
Zone 1 & 2		36.36	38.45	11.81
Random 500m²		30.39	33.54	5.89

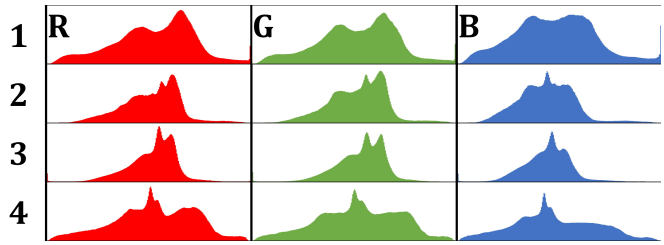

Fig. 7. Red-Green-Blue color histogram of S3DIS zone 1 with 1) raw colors, 2) MSE loss only 3) MSE + distance regularization 4) MSE + variance regularization.

TABLE V
COMPARATIVE STUDY ON COLORING REGULARISATION; RESULTS IN MIOU.

Coloring Method \ Color during testing	With	Without
Original color	31.73	9.97
Without color	-	*33.51*
MSE loss	**33.99**	**17.80**
MSE loss + dist. reg.	33.34	15.34
MSE loss + var. reg.	33.08	17.35

color information and justifies the importance of a coloring which focuses on important, specific features, to improve the spatial awareness of said networks. Training a segmentation network on a colored zone 1 (Tab. V) confirms this hypothesis: performance obtained is superior to both a network trained only on spatial information ("Without color") and a network trained with original colors.

Due to the nature of the loss (Eq. (1)) used for training the coloring network and the large quantity of training data, the extremes of the color scale are not well represented ([0;30] and [220;255]) after coloring (Fig. 7). The two proposed regularization strategies (Eq. (2) and Eq. (3)) are tested by training the segmentation network on differently colored zone 1 of S3DIS (Tab. V). Forcing the full use of the color scale creates chromatic aberrations, uniform coloring on part which should be distinguished and a blurry coloring (Fig. 8). A full use of the color scale is harmful for segmentation performance compared to a sparse but meaningful coloring.

Finally, another advantage of the coloring method is the gain in robustness towards color loss with a reduction in performance of 48%(MSE) instead of 69%(Raw) (Tab. V). Its consistency with each regularisation confirms the MSE loss role in creating color information which reduces over-fitting.

E. Ablation study

To prove the necessity of each component, an ablation study of the method is carried out. The first ablation is the use of random surface sampling (RSS) instead of Virtual Laser Sampling (VLS). The long-range virtual laser is used for

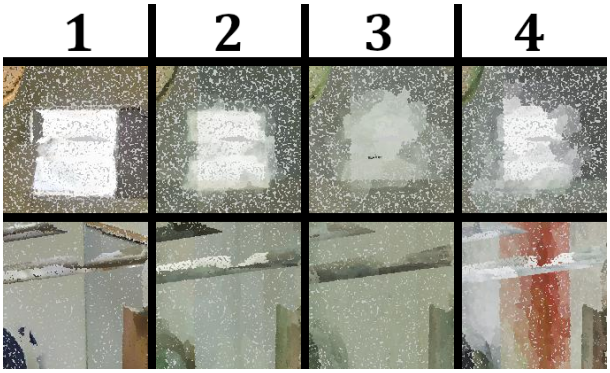

Fig. 8. Observable defects of coloring on S3DIS zone 1 with 1) raw colors, 2) MSE loss only 3) MSE + distance regularization 4) MSE + variance regularization.

sampling in the VLS case. The second and third ablations are using colorless (CC) and randomly colored (RC) synthetic data (i.e. each point color is determined randomly from 255^3 possible colors) instead of colored ones. The third ablation test is necessary to determine if the coloring method only helps to reduce over-reliance on color information during segmentation, as in such case, improvement to segmentation brought by random colors should be similar to the devised coloring method. Finally, when specified the SceneNet RGB-D scene creation method is used (SNC) in lieu of the devised method as a fourth ablation. The same experimental conditions as Sec. IV-C are used and the best of three is reported Table VI.

Differences in the results depending on color (Tab. VI) confirm the importance of coloring showed in Sec. IV-D. Training on colorless data (CC) reduces performance whether VLS sampling or random sampling is used. The use of random colors (RC) does not lead to a remarkable increase of performance, if it could reduce over-fitting, it does not improve the training effectiveness. This confirms the role of coloring in injecting additional information learned on unlabeled data in the synthetic ones. Moreover, it helps the network in using color only when the information conveyed is coherent with position and normal information such as color variation due to lighting condition (Fig. 9). The coloring mostly helps in the case of less represented classes (chair, clutter, door, sofa and window). As those are the classes whose synthetic generation is best handled (ie. fidelity of 3D models), coloring is thus most efficient when the geometrical domain-shift is moderate (see Fig. 9 third row : no hallway layout were used for scene generation, the gain observed are minimal in this case).

Changing the sampling method (RSS) also produces a significant drop in performance, with counter-productive results. This is due to the increase in the domain shift provoked by such sampling method. If Virtual Laser Scanning helps by producing occlusions and only capturing what is visible from certain points of view, random sampling creates points for every surface. When using synthetic data, walls can be modeled by thin rectangular cuboid shapes which will in turn create two surfaces of points. This duplication of certain surfaces, not present in real data, will interfere with the segmentation network understanding of the scene. Surface duplication of

TABLE VI
DIFFERENCES OBTAINED BY CHANGING THE SAMPLING METHOD AND REMOVING THE COLORING OF SYNTHETIC SCENES.

Sampling and Coloring	OA	mIoU	mIoU*	beam	board	book.	ceiling	chair	clut.	door	floor	table	wall	column	sofa	window
VLS colored	**71,1**	**34,4**	**43.4**	0,1	21,9	44,3	62,0	**65,0**	**37,5**	**35,5**	57,5	54,2	**70,6**	3,8	**9,5**	**19,0**
VLS + CC	69,1	33,1	41.7	0,2	**23,3**	**46,2**	59,7	64,9	36,9	28,7	54,1	56,5	67,9	4,6	8,0	13,0
VLS + RC	70,9	33,4	42.0	0,2	19,7	43,5	**63,1**	63,0	37,3	30,7	**60,6**	**58,2**	68,9	4,6	4,9	12,5
SNC + VLS + CC	69.2	31.7	39.8	0,1	10,0	46,0	62,4	60,5	36,6	26,5	54,6	57,5	67,7	**5.0**	6,5	9,9
RSS colored	69,3	31,6	39.8	0,0	13,0	44,4	61,7	61,1	37,1	28,1	53,7	50,7	68,8	3,9	4,4	15,3
RSS + CC	68,1	30,9	38.9	0,2	14,0	44,4	61,7	59,3	36,3	27,3	51,6	51,2	66,9	4,6	6,0	8,7
Zone 1 Only	67,6	31,7	39.9	0,1	18,7	44,2	60,7	61,5	37,1	27,7	46,4	53,6	68,0	**5,0**	7,7	13,7

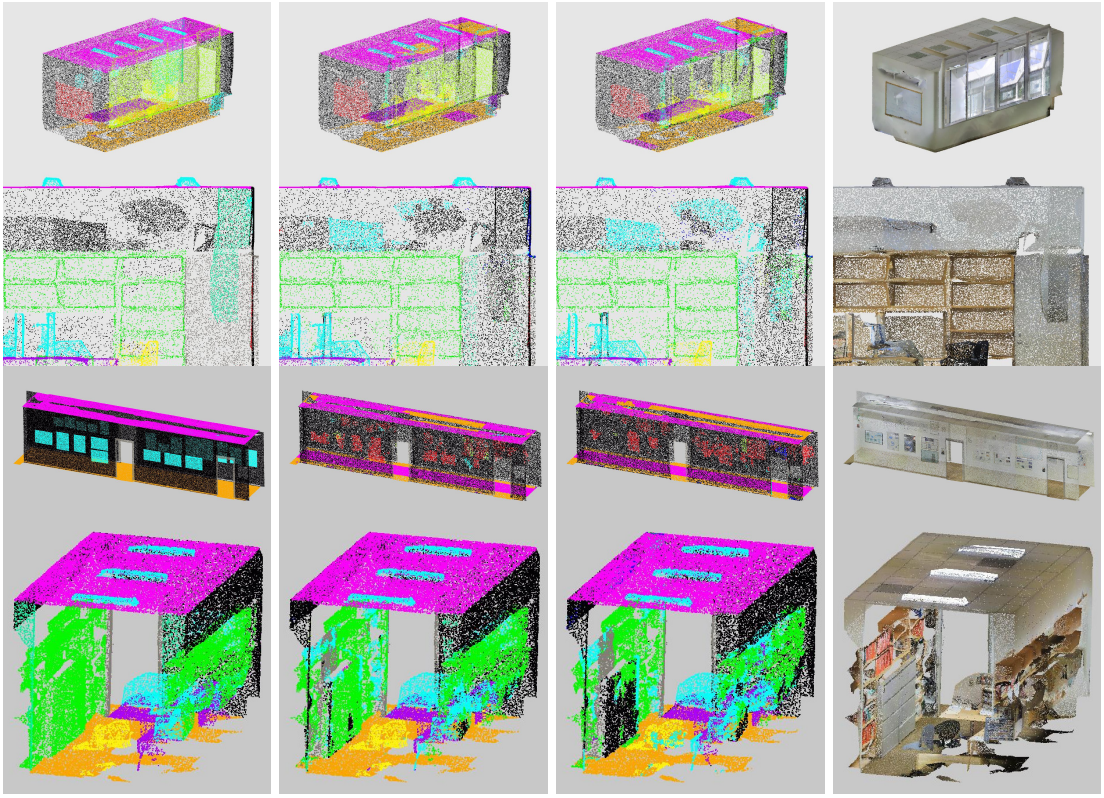

Fig. 9. Semantic segmentation tests results obtained after training with synthetic data sampled with VLS, from left to right : truth, colored synthetic data, colorless synthetic data, raw scene.

wall, ceiling and floor impedes the understanding of the general layout of the scene. When applied to objects, this effect impedes the network in learning two things : the same object can be cut by occlusion (a chair and its feet) and parts of an object can be absent depending on the acquisition process.

These results show that every component of the method helps in reducing the domain shift problem and should be used jointly when possible.

V. CONCLUSION

In this work, different approaches for reducing the synthetic data augmentation domain-shift problem by operating directly on the data, were presented. Such methods are compatible and complementary with those operating on the trained segmentation network. Moreover, even in the case were the synthetic data alone does not improve segmentation performance, the presented methods allow for an improvement in segmentation. In particular, virtual laser sampling can be used with any scene mesh to create a point cloud with realistic artifacts, noise and occlusions (Sec. III-B). Coloring can be used to inject information from unlabeled real world data into colorless synthetic point clouds, allowing for a performance gain at a low man-hour cost (Sec. III-C). Finally, the proposed scene generation method allows to quickly generate synthetic data and requires little human supervision (Sec. IV-B).

The main potential for going beyond this work lies in the synthetic data generation process. Even if it is fairly general and could be applied to other domains, it is the ultimate source of domain distance between synthetic data and real ones. To create better synthetic data, efforts targeting specifically the domain of application should be made: for example in the experiment carried out in this article, the mesh models used to create the synthetic scene as well as the produced scene

structure could have been refined to more precisely target the S3DIS dataset. Further work could be oriented to evaluate the impact of scene generators with domain-specific knowledge and domain targeted object models.

With respect to the sampling and coloring methods, their suitability to other applications domains will be investigated. First results bring confidence that these two methods could be transposed with little effort and still enable improvement to segmentation results. However, the virtual laser positioning could require modifications and the use of expert-knowledge in the application domain. The proposed positioning method has only been tested with interior scenes with simple geometry, its efficiency needing to be evaluated in more challenging environments.

Finally, GAN coloring, which is known to produce realistic results, could be compared to the proposed coloring method in terms of semantic segmentation improvement.

REFERENCES

[1] Iro Armeni, Sasha Sax, Amir R. Zamir, and Silvio Savarese. Joint 2D-3D-Semantic Data for Indoor Scene Understanding. *arXiv:1702.01105 [cs]*, 2017. 5

[2] Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3D Semantic Parsing of Large-Scale Indoor Spaces. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 4, 5

[3] Alexandre Boulch, Joris Guerry, Bertrand Le Saux, and Nicolas Audebert. SnapNet: 3D point cloud semantic labeling with 2D deep segmentation networks. *Computers and Graphics*, 71, 2018. 2

[4] Xu Cao and Katashi Nagao. Point Cloud Colorization Based on Densely Annotated 3D Shape Dataset. In *MultiMedia Modeling*, volume 11295, 2019. 3

[5] Xu Cao, Weimin Wang, Katashi Nagao, and Ryosuke Nakamura. PSNet: A Style Transfer Network for Point Cloud Stylization on Geometry and Color. In *IEEE Winter Conference on Applications of Computer Vision*, 2020. 3, 4

[6] F. M. Carlucci, P. Russo, and B. Caputo. (DE)²CO: Deep Depth Colorization. *IEEE Robotics and Automation Letters*, 2018. 3

[7] Romain Cazorla, Line Poinel, Panagiotis Papadakis, and Cédric Buche. Bottleneck identification to semantic segmentation of industrial 3d point cloud scene via deep learning. In *International Joint Conference on Artificial Intelligence*, 2021. 2

[8] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An Information-Rich 3D Model Repository. Technical report, Stanford University, Princeton University, Toyota Technological Institute at Chicago, 2015. 2, 5

[9] Thomas Chaton, Nicolas Chaulet, Sofiane Horache, and Loic Landrieu. Torch-Points3D: A modular multi-task framework for reproducible deep learning on 3D point clouds. In *International Conference on 3D Vision*, 2020. 5

[10] Yulan Guo, Hanyun Wang, Qingyong Hu, Hao Liu, Li Liu, and Mohammed Bennamoun. Deep Learning for 3D Point Clouds: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 1, 2

[11] Ankur Handa, Viorica Patraucean, Vijay Badrinarayanan, Simon Stent, and Roberto Cipolla. Understanding RealWorld Indoor Scenes with Synthetic Data. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 5

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Conference on Computer Vision and Pattern Recognition*, 2016. 4

[13] Muhammad Imad, Oualid Doukhi, and Deok-Jin Lee. Transfer Learning Based Semantic Segmentation for 3D Object Detection from Point Cloud. *Sensors*, 21(12), 2021. 1

[14] Li Jiang, Shaoshuai Shi, Zhuotao Tian, Xin Lai, Shu Liu, Chi-Wing Fu, and Jiaya Jia. Guided Point Contrastive Learning for Semi-Supervised Point Cloud Semantic Segmentation. In *IEEE/CVF International Conference on Computer Vision*, 2021. 1

[15] Jitao Liu, Songmin Dai, and Xiaoqiang Li. PCCN:POINT Cloud Colorization Network. In *IEEE International Conference on Image Processing*, 2019. 3

[16] John McCormac, Ankur Handa, Stefan Leutenegger, and Andrew J. Davison. SceneNet RGB-D: Can 5M Synthetic Images Beat Generic ImageNet Pre-training on Indoor Segmentation? In *IEEE International Conference on Computer Vision*, 2017. 2, 3

[17] Hsien-Yu Meng, Lin Gao, YuKun Lai, and Dinesh Manocha. VV-Net: Voxel VAE Net with Group Convolutions for Point Cloud Segmentation. In *IEEE/CVF International Conference on Computer Vision*, 2019. 2

[18] Florian Noichl, Alex Braun, and Andre Borrmann. "BIM-to-Scan" for Scan-to-BIM: Generating Realistic Synthetic Ground Truth Point Clouds based on Industrial 3D Models. In *European Conference on Computing in Construction*, 2021. 2, 3

[19] Charles Ruizhongtai Qi, Hao Su, Mo Kaichun, and Leonidas J Guibas. PointNet : Deep Learning on Point Sets for 3D Classification and Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2, 4

[20] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In *International Conference on Neural Information Processing Systems*, 2017. 4

[21] Swami Sankaranarayanan, Yogesh Balaji, Arpit Jain, Ser Nam Lim, and Rama Chellappa. Learning from Synthetic Data: Addressing Domain Shift for Semantic Segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018. 1, 2

[22] Jonathan Sauder and Bjarne Sievers. Self-Supervised Deep Learning on Point Clouds by Reconstructing Space. In *Advances in Neural Information Processing Systems*, volume 32, 2019. 1

[23] Takayuki Shinohara, Haoyi Xiu, and Masashi Matsuoka. Point2color: 3D Point Cloud Colorization Using a Conditional Generative Network and Differentiable Rendering for Airborne LiDAR. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2021. 3

[24] Alessandro Tasora, Radu Serban, Hammad Mazhar, Arman Pazouki, Daniel Melanz, Jonathan Fleischmann, Michael Taylor, Hiroyuki Sugiyama, and Dan Negrut. Chrono: An Open Source Multi-physics Dynamics Engine. In *High Performance Computing in Science and Engineering*, 2016. 3

[25] Hugues Thomas, Charles R. Qi, Jean Emmanuel Deschaud, Beatriz Marcotegui, Francois Goulette, and Leonidas Guibas. KPConv: Flexible and deformable convolution for point clouds. In *IEEE International Conference on Computer Vision*, 2019. 2, 4, 5

[26] Lei Wang, Yuchun Huang, Yaolin Hou, Shenman Zhang, and Jie Shan. Graph Attention Convolution for Point Cloud Semantic Segmentation. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019. 2

[27] Lukas Winiwarter, Alberto Manuel Esmorís Pena, Hannah Weiser, Katharina Anders, Jorge Martínez Sánchez, Mark Searle, and Bernhard Höfle. Virtual laser scanning with HELIOS++: A novel take on ray tracing-based simulation of topographic full-waveform 3D laser scanning. *Remote Sensing of Environment*, 269, 2022. 2, 3

[28] Bichen Wu, Alvin Wan, Xiangyu Yue, and Kurt Keutzer. SqueezeSeg: Convolutional Neural Nets with Recurrent CRF for Real-Time Road-Object Segmentation from 3D LiDAR Point Cloud. In *IEEE International Conference on Robotics and Automation*, 2018. 2, 3

[29] Bichen Wu, Xuanyu Zhou, Sicheng Zhao, Xiangyu Yue, and Kurt Keutzer. SqueezeSegV2: Improved Model Structure and Unsupervised Domain Adaptation for Road-Object Segmentation from a LiDAR Point Cloud. In *International Conference on Robotics and Automation*, 2019. 1, 2, 3

[30] Yuxing Xie, Jiaojiao Tian, and Xiao Xiang Zhu. Linking Points With Labels in 3D: A Review of Point Cloud Semantic Segmentation. *IEEE Geoscience and remote sensing magazine*, 8, 2019. 1

[31] Zaiwei Zhang, Rohit Girdhar, Armand Joulin, and Ishan Misra. Self-Supervised Pretraining of 3D Features on Any Point-Cloud. In *IEEE/CVF International Conference on Computer Vision*, 2021. 1

[32] Na Zhao, Tat-Seng Chua, and Gim Hee Lee. Few-shot 3D Point Cloud Semantic Segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 1

[33] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3D ShapeNets: A deep representation for volumetric shapes. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015. 2