ChangeGAN: A Deep Network for Change Detection in Coarsely Registered Point Clouds

Balázs Nagy, Lóránt Kovács, and Csaba Benedek

***Abstract*—In this letter we introduce a novel change detection approach called *ChangeGAN* for coarsely registered point clouds in complex street-level urban environment. Our generative adversarial network-like (GAN) architecture compounds Siamese-style feature extraction, U-net-like use of multiscale features, and Spatial Trans-formation Network (STN) blocks for optimal transformation estimation. The input point clouds are represented by range images, which enables the use of 2D convolutional neural networks. The result is a pair of binary masks showing the change regions on each input range image, which can be backprojected to the input point clouds without loss of information. We have evaluated the proposed method on various challenging scenarios and we have shown its superiority against state-of-the-art change detection methods.**

***Index Terms*—Change detection, lidar, deep learning for visual perception, range sensing.**

I. Introduction

DUE to the increasing population density, the rapid development of smart city applications and autonomous vehicle technologies, growing demand is emerging for automatic public infrastructure monitoring and surveillance applications. Detecting possibly dangerous situations caused by e.g. missing traffic signs, faded road signs and damaged street furniture is crucial. Expensive and time-consuming efforts are required therefore by city management authorities to continuously analyze and compare multi-temporal recordings from large areas to find relevant environmental changes.

From the perspective of machine perception, this task can be formulated as a change detection (CD) problem. In video surveillance applications [1], [2] change detection is a standard approach for scene understanding by estimating the background regions and by comparing the incoming frames to this background model. Change detection is also a common task in many remote sensing (RS) applications, which require the extraction of the differences between aerial images, point clouds, or other measurement modalities [3], [4]. However, the vast majority of existing approaches assume that the compared image or point cloud frames are precisely registered since either the sensors are motionless or the accurate position and orientation parameters of the sensors are known at the time of each measurement.

Mobile and terrestrial Lidar sensors can obtain point cloud streams providing accurate 3D geometric information in the observed area. Lidar is used in autonomous driving applications supporting the scene understanding process, and it can also be part of the sensor arrays in ADAS systems of recent high-end cars. Since the number of vehicles equipped with Lidar sensors is rapidly increasing on the roads, one can utilize the tremendous amount of collected 3D data for scene analysis and complex street-level change detection. Besides, change detection between the recorded point clouds can improve virtual city reconstruction or Simultaneous Localization and Mapping (SLAM) algorithms [5].

Processing street-level point cloud streams is often a significantly more complex task than performing change detection in airborne images or Lidar scans. From a street-level point of view, one must expect a larger variety of object shapes and appearances, and more occlusion artifacts between the different objects due to smaller sensor-object distances. Also, the lack of accurate registration between the compared 3D terrestrial measurements may mean a crucial bottleneck for the whole process, for two different reasons: *First*, in a dense urban environment, GPS/GNSS-based accurate self-localization of the measurement platform is often not possible [6]. *Second*, the differences in viewpoints and density characteristics between the data samples captured from the considered scene segments may make automated point cloud registration algorithms less accurate [6].

In this paper, a deep neural network-based change detection approach is proposed, which can robustly extract changes between sparse point clouds obtained in a complex street-level environment. As a key feature, the proposed method does not require precise registration of the point cloud pairs. Based on our experiments, it can efficiently handle up to $1m$ translation and 10° rotation misalignment between the corresponding 3D point cloud frames.

Manuscript received April 14, 2021; accepted August 1, 2021. Date of publication August 18, 2021; date of current version September 2, 2021. This work was supported in part by the National Research Development and Innovation (NRDI) Office within the frameworks of the Autonomous Systems National Laboratory and the Artificial Intelligence National Laboratory programs, in part by the NRDI Grants K-120233 and 2018-2.1.3-EUREKA-2018-00032, in part by the Széchenyi 2020 Program under Grant EFOP-3.6.3-VEKOP-16-2017-00002, and in part by the ÚNKP-20-3, and ÚNKP-20-4 New National Excellence Program of the Ministry for Innovation, and Technology. *(Balázs Nagy and Lóránt Kovács are co-first authors.)* This letter was recommended for publication by Associate Editor Prof. Maani Ghaffari and Dr. Cesar Cadena Lerma upon evaluation of the reviewers' comments. *(Corresponding author: Lóránt Kovács.)*

The authors are with the Institute for Computer Science and Control (SZTAKI), Eötvös Loránd Research Network, 1111 Budapest, Hungary, and also with the Péter Pázmány Catholic University, 1083 Budapest, Hungary (e-mail: nagy.balazs@sztaki.hu; kovacs.lorant@sztaki.hu; benedek.csaba@sztaki.hu).

Digital Object Identifier 10.1109/LRA.2021.3105721

II. Related Works

As one of the most fundamental problems in multitemporal sensor data analysis, change detection has had a vast bibliography in the last decade. Besides methods working on remote sensing images, several change detection techniques deal with *terrestrial* measurements, where the sensor is facing towards the horizon and is located on or near the ground. In these tasks optical cameras [7] and rotating multi-beam Lidars [8] are frequently used, solving problems related to surveillance, map construction, or SLAM algorithms [9].

A. Prior Approaches

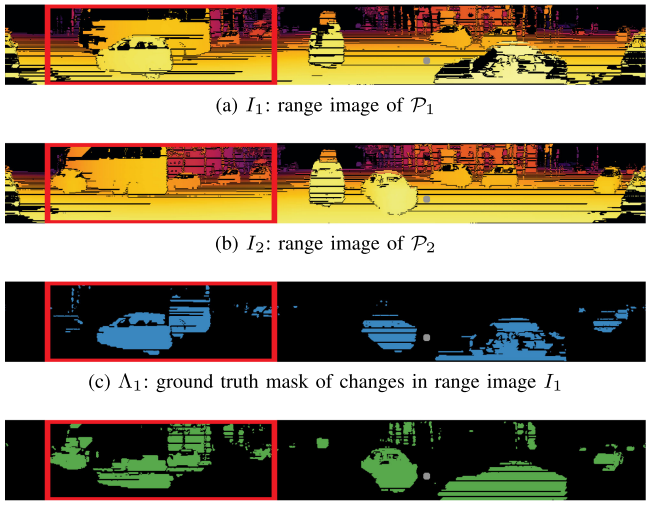
We can categorize the related works based on the applied methodology they use for change detection. Many approaches are based on *handcrafted features*, such as a set of pixel- and object-level descriptors [10], occupancy grids [11], volumetric features, and point distribution histograms [9], but they all need preliminarily registered inputs. Only a few feature-based techniques deal with compensating small misregistration effects, such as [12], where terrestrial images and point clouds are fused to perform change detection.

Neural network-based change detection techniques can handle in general more robustly the variances originated from viewpoint differences, most frequently using Siamese network architectures. However, prior approaches solely focus here on visual change detection problems in aerial [13] or street-view [7], [14] optical image pairs, and this task is yet to be solved for real Lidar point cloud-based change detection problems. A new method for detecting structural changes from city images is described in [15]. It creates 3D point Clouds using Structure-from-Motion (SfM) from the images and uses a deep-learning based registration on the 3D clouds.

B. Registration Issues

Most of the aforementioned methods require that the compared measurements are either recorded from a static platform, or they can be accurately registered into a joint coordinate system by using external navigation sensors, and/or robust image/point cloud matching algorithms. The later registration step is critical for real-world 3D perception problems, since the recorded 3D point clouds often have strongly inhomogeneous density, and the blobs of the scanned street-level objects are sparse and incomplete due to occlusions and the availability of particular scanning directions only. Under such challenging circumstances, conventional point-to-point, patch-to-patch, or point-to-patch correspondence-based registration strategies often fail [16].

To our best knowledge, this paper presents the first approach to solve the change detection problem among sparse, coarsely registered terrestrial point clouds, without needing an explicit fine registration step. Utilizing the STN layer, the model can automatically handle errors of coarse registration. Our proposed deep learning-based method can extract and combine various low-level and high-level features throughout the convolutional layers, and it can learn semantic similarities between the point clouds, leading to its capability of detecting changes without prior registration. A clear difference between the proposed change detection method and the state-of-the-art is the adversarial training strategy which has a regularization effect, especially on limited data. The other main difference is the built-in spatial transformer network yielding the proposed model to be able to learn and handle coarse registration errors.



(a) I_1: range image of $\mathcal{P}_1$

(b) I_2: range image of $\mathcal{P}_2$

(c) Λ_1: ground truth mask of changes in range image I_1

(d) Λ_2: ground truth mask of changes in range image I_2

Fig. 1. Input data representation. (a),(b): range images I_1, I_2 from a pair of coarsely registered point clouds $\mathcal{P}_1$ and $\mathcal{P}_2$. (c),(d): binary ground truth change masks Λ_1, Λ_2 for the range images I_1 and I_2, respectively. The *red rectangle* marks the region also displayed in Fig. 6.

III. Proposed Method

Several Lidar devices, such as the Rotating multi-beam (RMB) sensors manufactured by Velodyne and Ouster, can provide high frame-rate point cloud streams containing accurate, but relatively sparse 3D geometric information from the environment. These point clouds can be used for infrastructure monitoring, urban planning [17], and SLAM [5].

The goal of our proposed solution is to extract changes between two coarsely registered sparse Lidar point clouds, $\mathcal{P}_1$ and $\mathcal{P}_2$. To formally define our change detection task, several considerations should be taken. *First*, both input point clouds may contain various dynamic or static objects, which are not present in the other measurement sample. *Second*, due to the lack of registration, we cannot use a single common voxel grid for marking the locations of changes between the two point clouds. Instead, using a $\mu(.)$ point labeling process, we separately mark each point $p \in \mathcal{P}_1 \cup \mathcal{P}_2$ as changed ($\mu(p) = \mathrm{ch}$) or unchanged background ($\mu(p) = \mathrm{bg}$) point, respectively. We label a point $p_1 \in \mathcal{P}_1$ as changed if the surface patch represented by point p_1 in $\mathcal{P}_1$ is not present (changed or occluded) in point cloud $\mathcal{P}_2$ (the label of a point $p_2 \in \mathcal{P}_2$ is similarly defined). Results of the proposed classification approach for a sample 3D point cloud pair are demonstrated in Fig. 4.

A. Range Image Representation

Our proposed solution extracts changes between two coarsely registered Lidar point clouds in the range image domain. For

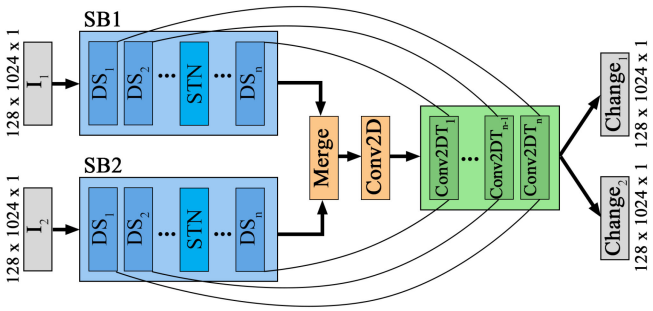


Fig. 2. Proposed *ChangeGAN* architecture. Notations of components: SB1, SB2: Siamese branches, DS: downsampling, STN: spatial transformer network, Conv2DT: transposed 2D convolution.

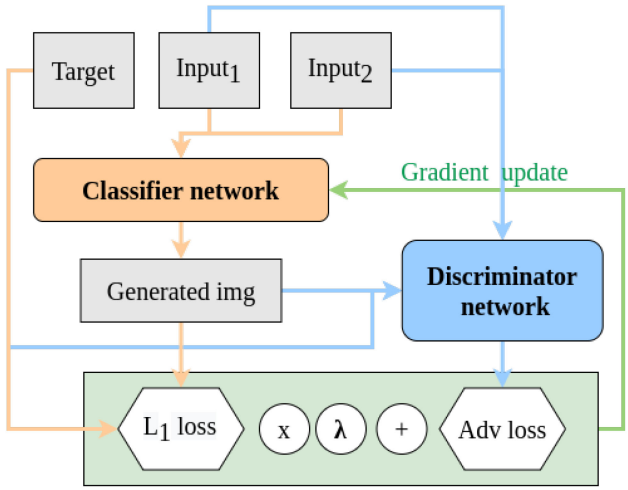


Fig. 3. Proposed adversarial training strategy of the *ChangeGAN* architecture.

example, creating a range image from a rotating multi-beam (RMB) Lidar sensor's point stream is straightforward [18] as its laser emitter and receiver sensors are vertically aligned, thus every measured point has a predefined vertical position in the image, while consecutive firings of the laser beams define their horizontal positions. Geometrically, this mapping is equivalent to transforming the representation of the point cloud from the 3D Descartes to a spherical polar coordinate system, where the polar direction and azimuth angles correspond to the horizontal and vertical pixel coordinates, and the distance is encoded in the corresponding pixel's 'intensity' value. Note that range image mapping can also be implemented for other (non-RMB) Lidar technologies, such as for Livox sensors. Using appropriate image resolution the conversion of the point clouds to 2D range images is reversible, without causing information loss. Besides providing a compact data representation, using the range images makes it also possible to adopt 2D convolution operations by the used neural network architectures.

The proposed deep learning approach takes as input two coarsely registered 3D point clouds $\mathcal{P}_1$ and $\mathcal{P}_2$ represented by range images I_1 and I_2, respectively (shown in Fig. 1(a), and 1(b)) to identify changes. Our architecture assumes that the images I_1 and I_2 are defined over the same pixel lattice S, and have the same spatial *height (h)*, *width (w)* dimensions.

Usually, change detection algorithms working on multitemporal image pairs [7] explicitly define a test and a reference sample, and changes are interpreted from the perspective of the reference data: the resulting change mask marks the image regions which are changed in the test image compared to the reference one. However, this approach cannot be adopted in our case. It is not relevant to assign a single binary change/background label to the pixels of the joint lattice S of the range images, as they may represent different scene locations in the two input point clouds. For this reason, we represent the change map by a two-channel mask image over S, so that to each pixel $s \in S$ we assign two binary labels $\Lambda_1(s)$ and $\Lambda_2(s)$. Following our change definition used earlier in 3D, for $i \in \{1, 2\}$, $\Lambda_i(s) = \text{ch}$ encodes that the 3D point $p_i \in \mathcal{P}_i$ projected to pixel s should be marked as change in the original 3D point cloud domain of $\mathcal{P}_i$, i.e. $\mu(p_i) = \text{ch}$ (see Fig. 1(c), and 1(d)).

Next, our change detection task can be reformulated in the following way: our network extracts similar features from the range images I_1 and I_2, then it searches for the high correlation between the features, and finally, it maps the correlated features to two binary change mask channels Λ_1 and Λ_2, having the same size as the input range images.

B. *ChangeGAN Architecture*

For our purpose, we propose a new generative adversarial neural network-like architecture, more specifically a discriminative method, with an additional adversarial discriminator as a regularizer, called *ChangeGAN*, which is shown in Fig. 2.

Since the main goal is to find meaningful correspondences between the input range images I_1 and I_2, we have adopted a Siamese style [19] architecture to extract relevant features from the input range image pairs. The Siamese architecture is designed to share the weight parameters across multiple branches allowing us to extract similar features from the inputs and to decrease the memory usage and training time. Each branch of the Siamese network consists of fully convolutional down-sampling (DS) blocks. The first layer of the DS block is a 2D convolutional layer with a stride of 2 which has a 2-factor down-sampling effect along the spatial dimensions. This step is followed by using a batch normalization layer, and finally, we activate the output of the DS block using a leaky ReLU function. Next, we concatenate the outputs of the Siamese branches for all feature channels, and we apply a 1×1 convolutional layer to aggregate the merged features. The second part of the proposed model contains a series of transposed convolutional layers to up-sample the signal from the lower-dimensional feature space to the original size of the 2D input images. Finally, a 1×1 convolutional layer, activated with a sigmoid function, generates the two binary change maps Λ_1 and Λ_2. To regularize the network and prevent over-fitting we use the Dropout technique after the first two transposed convolutional layers. To improve the change detection result we have adapted an idea from U-net [20] by adding higher resolution features from the DS blocks to the corresponding transposed convolutional layers.

The branches of the Siamese network can extract similar features from the inputs. In our case, as the point clouds are coarsely

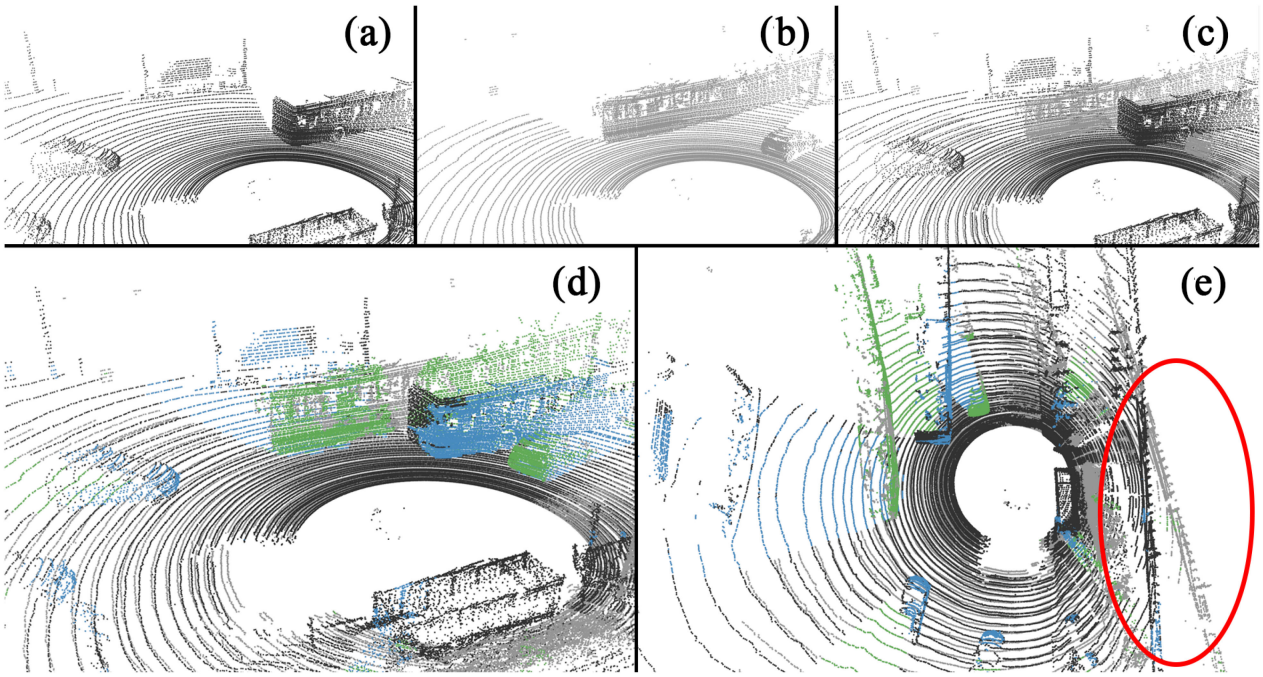


Fig. 4. Changes detected by *ChangeGAN* for a coarsely registered point cloud pair. (a) and (b) show the two input point clouds, (c) displays the coarsely registered input point clouds in a common coordinate system. (d),(e) present the change detection results: blue and green colored points represent the objects marked as changes in the first and second point cloud, respectively. The red ellipse draws attention to the global alignment difference between the two coarsely registered point clouds.

registered, the same regions of the input range images might not be correlated with each other. To achieve more accurate feature matching we have added Spatial Transformation Network (STN) blocks [21] for both Siamese branches (see Fig. 2). STN can learn an optimal affine transformation between the input feature maps to reduce the spatial registration error between the input range images. Furthermore, STN dynamically transforms the inputs, also yielding an advantageous augmentation effect.

C. Training ChangeGAN

A competitive classifier - discriminator-based adversarial training was implemented for the *ChangeGAN* network.

The *classifier* network is responsible for learning and predicting the changes between the range image pairs. In each training epoch, the classifier model is trained on a batch of data. The actual state of the classifier is used to predict validation data which is fed to the discriminator model.

The *discriminator* network is a fully convolutional network that classifies the output of the classifier network. The discriminator model divides the image into patches and decides for each patch whether the predicted change region is real or fake. During training, the discriminator network forces the classifier model to create better and better change predictions, until the discriminator cannot decide about the genuineness of the prediction.

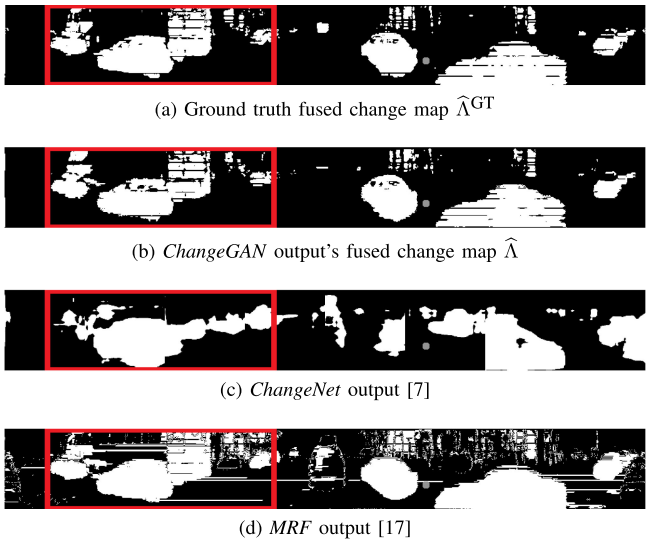
Fig. 3 demonstrates the proposed adversarial training strategy. We calculate the L1 Loss (L_{L1}) as the mean absolute error between the generated image and the target image, and we define the Adversarial Loss (L_{Adv}), which is a sigmoid cross-entropy loss of the feature map generated by the discriminator and an array of ones. The final loss function of the method (L) is the weighted combination of the Adversarial Loss and the L1 Loss: $L = L_{\text{Adv}} + \lambda * L_{L1}$. Based on our experiments we set $\lambda = 300$.

Both the classifier and the discriminator part of the GAN-like architecture were optimized by the Adam optimizer and the learning rate was set to 10^{-5}. We have trained the model on 300 epochs which takes almost two days. At each training epoch, we have updated the weights of both the classifier and the discriminator ones.

We note here, that the ChangeGAN method can be trained without the Adversarial Loss (L_{Adv}), relying only on L1 loss. In our preliminary experiments, we followed this simpler approach, which was able to predict some change regions, but the results were notably ambiguous. To increase the generalization ability, we applied the adversarial training strategy in the proposed final model.

D. Change Detection Dataset

Considering that the main purpose of the presented *ChangeGAN* method is to extract changes from coarsely registered point clouds, for model training and evaluation we need a large, annotated set of point cloud pairs collected in the same area with various spatial offsets and rotation differences. Following our change definition in Section III, the annotation should accurately mark the point cloud regions of objects or scene segments that appear only in the first frame, only in the



(a) Ground truth fused change map $\widehat{\Lambda}^{\mathrm{GT}}$

(b) *ChangeGAN* output's fused change map $\widehat{\Lambda}$

(c) *ChangeNet* output [7]

(d) *MRF* output [17]

Fig. 5. Predicted change masks by the different methods on input data shown in Fig 1. Red rectangles: region shown in Fig. 6.

second frame, or which ones are unchanged thus observable in both frames (see Fig. 4 and 6).

Since the available point cloud benchmark sets cannot be used for this purpose, we have created a new Lidar-based urban dataset called *Change3D*[1]. Our measurements were recorded in the downtown of Budapest, Hungary on two different days by driving a car with a Velodyne HDL-64 rotating multi-beam Lidar attached to its roof. To our best knowledge, this *Change3D* dataset is the largest point cloud dataset for change detection, which contains both registered and coarsely registered point cloud pairs.

1) Ground Truth Creation Approach: Since manual annotation of changes between 3D point clouds is very challenging and time-consuming, we proposed a semi-automatic method using simulated registration errors to create ground truth (GT) for our change detection approach. To ensure the accuracy of the GT, we performed the change labeling for registered point cloud pairs captured from the same sensor position and orientation, then we randomly transformed the reference positions and orientations of the second frames yielding a large set of accurately labeled coarsely registered point cloud pairs. Thereafter, this set has been divided into disjunct training and test sets which could be used to train and quantitatively evaluate the proposed method.

The remaining parts of the collected data including originally unregistered point cloud pairs have been used for qualitative analysis through visual validation (see for example Fig. 4.) of the model performance.

2) Core Data Creation for GT Annotation: We selected 50 different locations during the test drive when the measurement platform was motionless for a period: it was stopped by traffic lights, crossroads, zebra crossings, parking situations, etc. These locations were taken both from narrow streets from the downtown and wide, large junctions as well. At each location, we took 100 recorded point clouds, and then we randomly selected 400 point cloud pairs among them, obtaining for the 50 locations a total number of 20000 point cloud pairs on which the training set was based. The test set is based on 2000 point cloud pairs, which were selected similarly, but in terms of locations and recording time stamps, the test samples were completely separated from the training data.

In these recordings, the differences among the point clouds were only caused by the moving dynamic objects such as vehicles and pedestrians. Alongside the exploitation of real object motion and occlusion effects, some further artificial changes have been synthesized by manually adding and deleting various street furniture elements to selected point cloud scenes. Also, we segmented the point clouds roughly to planes [22], and randomly deleted some selected 2D rectangular segments.

3) Semi-Automatic Change Extraction: Since the above-discussed frame pairs are taken in the same global coordinate system, they can be considered as *registered*. Their ground truth (GT) change annotation could be efficiently created in a semi-automatic way: A high-resolution 3D voxel map was built on a given pair of point clouds. The voxel size defines the resolution of the change annotation. The length of the change annotation cube was set to 0.1 m in all three dimensions. All voxels were marked as changed if 90% of the 3D points in the given voxel belonged to only one of the point clouds. Thereafter minor observable errors were manually eliminated by a user-friendly point cloud annotation tool. Finally, in both point clouds, all points belonging to *changed voxels* received a $\mu^{\mathrm{GT}}(p) = \mathrm{ch\ GT}$ labels, while the remaining points were assigned to $\mu^{\mathrm{GT}}(p) = \mathrm{bg}$ labels.

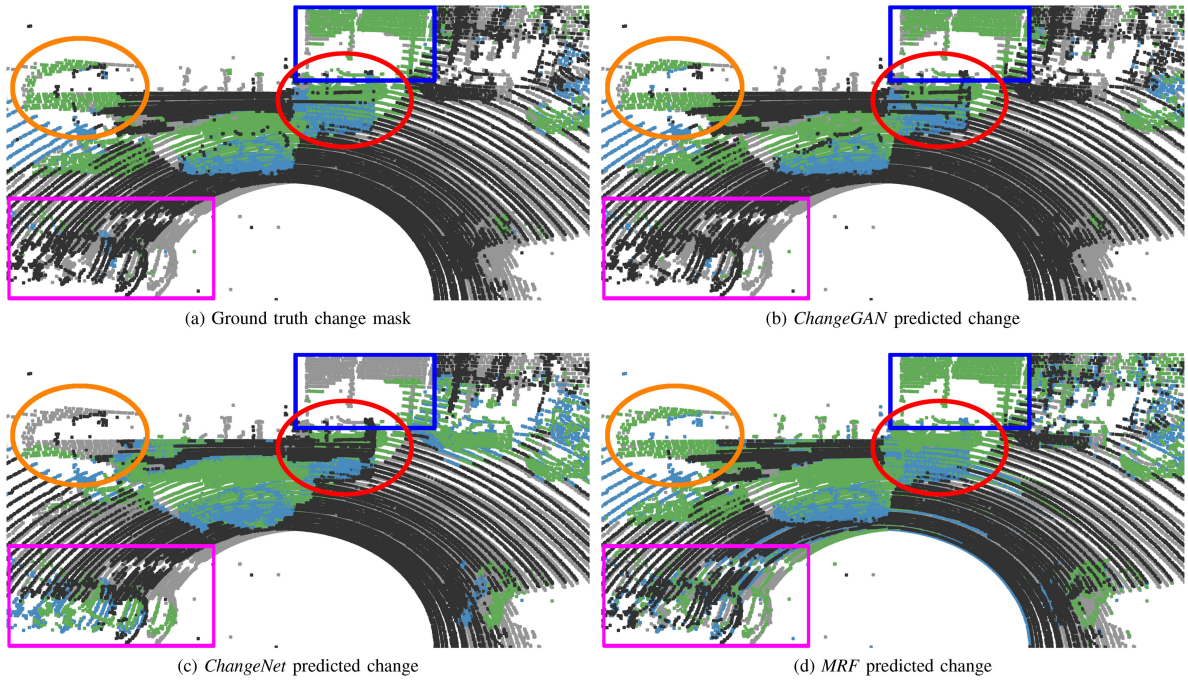
4) Registration Offset: To simulate the coarsely registered point cloud pairs requested by our *ChangeGAN* approach, we have applied randomly an up to $\pm 1m$ translation and an up to $\pm 10^{\circ}$ rotation transform around the z-axis for the second frame ($\mathcal{P}_2$) of each point cloud pair both in the training and test datasets. The $\mu^{\mathrm{GT}}(p)$ GT labels remained attached to the $p \in \mathcal{P}_2$ points and were transformed together with them.

5) Cloud Crop and Normalization: In the next step, all 3D points were removed from the point clouds, whose horizontal distances from the sensor were larger than $40m$, or their elevation values were greater than $5m$ above the ground level. This step yielded the capability of normalizing the point distances from the sensor between 0 and 1.

6) Range Image Creation and Change Map Projection: The transformed 3D point clouds were projected to 2D range images I_1, and I_2 as described in Section III-A (see Fig. 1). The Lidar's horizontal 360° field of view was mapped to 1024 pixels and the $5m$ vertical height of the cropped point cloud was mapped to 128 pixels, yielding that the size of the produced range image is 1024×128.

We note here that the Lidar sensor used in this experiment has 64 emitters yielding that the height of the original range images should be 64. However, to increase the learning capacity of the network we have doubled and interpolated the data among the height dimension since the 2D convolutional layers with a stride of 2 have a 2-factor down-sampling effect. Let us observe that the horizons of the range images are at similar positions in the two inputs due to the cropped height of the input point clouds.

[1]Dataset link: http://mplab.sztaki.hu/geocomp/Change3D.html



(a) Ground truth change mask (b) *ChangeGAN* predicted change

(c) *ChangeNet* predicted change (d) *MRF* predicted change

Fig. 6. Comparative results of the ground truth and the predicted changes by *ChangeGAN* and the reference techniques. Green and blue points mark changed regions in $\mathcal{P}_1$ and $\mathcal{P}_2$ respectively. Orange and red ellipses mark the detected front and back part of a bus travelling in the upper lane, meanwhile occluded by other cars. The blue square shows a building facade segment, which was occluded in $\mathcal{P}_2$. The magenta boxes highlight false positive changes of the reference methods confused by inaccurate registration.

Besides the range values, the $\mu^{\mathrm{GT}}(p)$ ground truth labels of the points were also projected to the Λ_1^{GT} and Λ_2^{GT} change masks, used for reference during training and evaluation of the proposed network.

IV. Experiments

We have trained and evaluated the proposed method using the new *Change3D* dataset (see Section III-D), which contains point cloud pairs recorded by a car-mounted RMB Lidar sensor at different times in dense city environments. For a selected coarsely registered point cloud pair, Fig. 4 shows the changes predicted by the proposed *ChangeGAN* model.

A. Reference Methods

To our best knowledge, we cannot find in the literature any reference methods focusing on change detection in *coarsely registered* terrestrial point clouds. However, since we reformulated the 3D change detection problem in the 2D range image domain, image-based methods tolerant of registration errors can also be taken into consideration for comparison.

As the *first* baseline, we have chosen the *ChangeNet* method [7], which is a recent approach for visual change detection, being able to detect and localize changes even if the scene has been captured at different lighting, view angle, and seasonal conditions. *ChangeNet* uses a *ResNet* backbone, working with fixed-size input images (224×224). Our created range images could not be given directly to this network, since their resolution (1024×128) and aspect ratio parameters are different. This issue was solved by splitting our range images into eight 128×128 parts, which were upscaled to the image size required by *ChangeNet*. We used the genuine and published implementation of the *ChangeNet* architecture, which was trained using our training data set described in Section III-D.

Our *second* reference method follows a voxel occupancy-based approach [17], where the detection accuracy and the ability to compensate minor registration errors depend on the chosen voxel resolution. As a core step of the algorithm, [17] applies a registration method between the point cloud pairs. For noise filtering and registration error elimination, a Markov Random Field (*MRF*) model is adopted which is defined in the range image domain [17].

Comparative results of the proposed method and the reference techniques for the point cloud pair of Fig. 1 are shown in Fig. 5 in range image representations.

Since neither the *ChangeNet* nor the *MRF* methods can distinguish changes by objects of the first and second images, for a direct comparison, we also binarized the output of *ChangeGAN* to get a fused change map $\widehat{\Lambda}$ where $\forall s \in S$: $\widehat{\Lambda}(s) = \max(\Lambda_1(s), \Lambda_2(s))$. The fused GT mask $\widehat{\Lambda}^{\mathrm{GT}}$ was similarly derived.

B. Quantitative Results

We evaluated the proposed *ChangeGAN* method and the two baseline techniques on our new *Change3D* benchmark set. The quantitative performance analysis was performed in the 2D range image domain, using the fused $\widehat{\Lambda}^{\mathrm{GT}}$ mask as a GT reference. To measure the similarity between the binary GT change mask and the binary change masks predicted by

TABLE I
PERFORMANCE COMPARISON OF THE PROPOSED *CHANGEGAN* METHOD TO *CHANGENET* [7] AND TO THE *MRF*-BASED REFERENCE APPROACH [17]

	ChangeGAN	ChangeNet	MRF-based
Accuracy	**0.93**	0.78	0.78
Precision	**0.83**	0.43	0.44
Recall	0.71	0.59	**0.88**
F1-score	**0.76**	0.48	0.58
IoU	**0.62**	0.42	0.32
Execution time [s]	0.06	**0.004**	0.51

the different methods, mean F1-score, Intersection over Union (IoU) were calculated alongside pixel-level precision, recall, and accuracy. The used metrics' definition follows a standard binary classification metrics [23].

The numerical evaluation results obtained by *MRF* [17], *ChangeNet* [7], and the proposed *ChangeGAN* methods over the 2000 range image pairs of the test dataset, are shown in Table I. As demonstrated, the *ChangeGAN* method outperforms both reference methods in terms of these performance factors, including the F1-score and IoU values.

The *MRF* [17] method is largely confused if the registration errors between the compared point clouds are significantly greater than the used voxel size. Such situations result in large numbers of falsely detected change-pixels, which fact yields on average very low precision result (0.44), although due to several accidental matches, the recall rate might be relatively high (0.88).

The measured low computational cost means a second strength of the proposed *ChangeGAN* approach, especially versus the *MRF* model, whose execution time is longer with one order of magnitude. Although *ChangeNet* is even faster than *ChangeGAN*, its performance is significantly weaker compared to the other two methods. Since the adversarial training strategy has a regularization effect [24], and the STN layer can handle coarse registration errors, the proposed *ChangeGAN* model can achieve better generalization ability and it outperforms the reference models on the independent test set. Note that in each case running speed was measured in seconds on a PC with an i8-8700 K CPU @3.7 GHz x12, 32 GB RAM, and a GeForce GTX 1080Ti.

C. Qualitative Results

For qualitative analysis, we backprojected the 2D binary change masks to the corresponding 3D point clouds and visually inspected the quality of the proposed change detection approach. During the investigations, we have observed similarly efficient performance for the remaining, originally unregistered point cloud pairs of the *Change3D* dataset, to the point cloud set with simulated registration errors which participated in the quantitative tests of Section IV-B.

For reasons of scope, we can only present here short discussions for two sample scenes displayed in Fig. 4 and 6.

Fig. 4 contains a busy road scenario, where different moving vehicles appear in the two point clouds. As shown, moving objects both from the first (blue color) and second (green) frames, are accurately detected despite the large global registration errors between the point clouds (highlighted by a red ellipse). Let us also observe that a change caused by a moving object in a given frame also implies a changed area in the other frame in its *shadow region*, which does not contain reflections due to occlusion. This phenomenon is a consequence of our change definitions, however, the shadow changes can be filtered out by geometric constraints, if they are not needed for a given application.

Fig. 6 displays another traffic situation, where the output of the proposed *ChangeGAN* technique can be compared to the manually verified Ground Truth and to the two reference methods in the 3D point cloud domain. As shown, our results accurately reflect our change concept defined in the paper, while the reference techniques cause multiple missing or false positive change regions. Since a bus travelling in the upper lane was partially occluded by other cars, only its frontal and the rear parts could be detected as changes. However, the *ChangeNet* model missed detecting its frontal region and a partially occluded facade segment. In addition, both reference methods detected false changes in the bottom left corner of the image, which were caused by the inaccurate registration. (Please find more details in the figure caption).

Finally, we note that our method has also successfully performed for frame pairs from the KITTI dataset [25], which were completely independent of our training process.

D. Robustness Analysis

To evaluate the performance dependency of the discussed methods on the translation and orientation differences between the compared point clouds, we generated two specific sample subsets within the new *Change3D* dataset. This experiment was based on 500 (originally registered) point cloud pairs, selected from the 2000 test sample pairs of the dataset.

For translation-dependency analysis, we used an offset domain of $[0.1, 1.0]$ meters, which was discretized using 10 equally spaced bins. For test set generation, we iterated through all the 500 point cloud pairs: For every sample, we chose for each translation bin $0.1 \leq t_i \leq 1.0$ $(i = 1 \dots 10)$ a random rotation value $-10^\circ \leq \alpha_i \leq 10^\circ$, and transformed the second cloud $\mathcal{P}_2$ using (t_i, α_i).

With this process, for each offset bin, we generated 500 coarsely registered point cloud pairs with known registration errors. In total, 10 subsets were created for the 10 offset bins, each one containing 500 samples.

Next, we run our proposed method and the reference techniques on this new set, and we calculated the mean F1-score [18], [26] value for each translation bin i, among samples having an offset parameter t_i. Fig. 7 displays with solid lines the average F1-scores in a function of various t_i values. The proposed method shows a graceful degradation by increased offsets, and even for a $t_i = 1$ meter offset, the quality of change detection is significantly better than the nearly constant low values provided by the reference approaches.

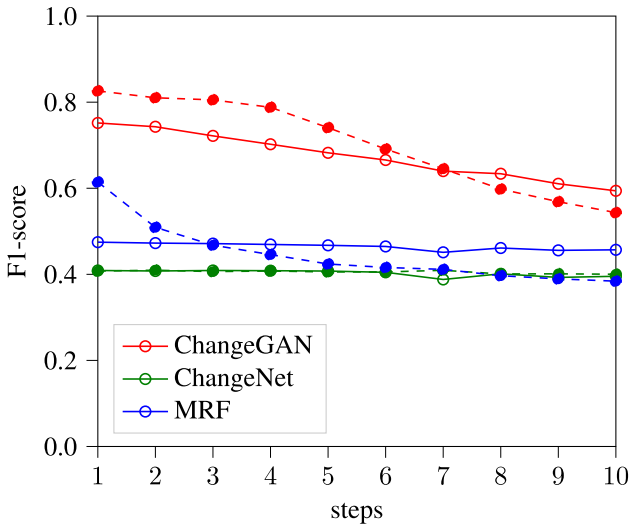


Fig. 7. Translation (solid lines) and rotation (dashed lines) dependency of the compared methods' performance (F1-score). Translation steps: $[0.1, 1.0]$ meters, rotation steps: $1^\circ : 10^\circ$.

For measuring the rotation-dependency of the models, we have performed a similar experiment: here we discretized the $-10^\circ \leq \alpha_i \leq 10^\circ$ rotation domain with 10 bins, and within each bin, we generated 500 sample pairs, with random translation values. Finally, we averaged the measured F1-scores within each rotation bin [18], [26]. Results shown in Fig. 7 with dashed lines confirm again the superiority of the proposed method against the tested references.

V. Conclusion

In this paper *ChangeGAN*, a novel, robust and quick change detection method was presented, which is capable of detecting differences between coarsely registered point cloud pairs. It has been shown that our approach outperforms in effectiveness both a state-of-the-art deep learning method (*ChangeNet*) trained on range images, and a 3D voxel-level, *MRF*-based change detection technique.

References

[1] C. Benedek, B. Gálai, B. Nagy, and Z. Jankó, "Lidar-based gait analysis and activity recognition in a 4D surveillance system," *IEEE Trans. Circuits Syst. Video Techn.*, vol. 28, no. 1, pp. 101–113, Jan. 2018.

[2] F. Oberti, L. Marcenaro, and C. S. Regazzoni, "Real-time change detection methods for video-surveillance systems with mobile camera," in *Proc. Eur. Signal Process. Conf.*, 2002, pp. 1–4.

[3] C. Benedek, X. Descombes, and J. Zerubia, "Building development monitoring in multitemporal remotely sensed image pairs with stochastic birth-death dynamics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 1, pp. 33–50, Jan. 2012.

[4] S. Ji, Y. Shen, M. Lu, and Y. Zhang, "Building instance change detection from large-scale aerial images using convolutional neural networks and simulated samples," *Remote Sens.*, vol. 11, no. 11, 2019, Art. no. 1343, [Online]. Available: https://www.mdpi.com/2072-4292/11/11/1343

[5] C.-C. Wang and C. Thorpe, "Simultaneous localization and mapping with detection and tracking of moving objects," in *Proc. Int. Conf. Robot. Automat.*, 2002, vol. 3, pp. 2918–2924.

[6] B. Nagy and C. Benedek, "Real-time point cloud alignment for vehicle localization in a high resolution 3D map," in *Proc. Eur. Conf. Comput. Vis. Workshops, Ser. Lecture Notes Comput. Sci.*, 2019, pp. 226–239.

[7] A. Varghese, J. Gubbi, A. Ramaswamy, and P. Balamuralidhar, "ChangeNet: A deep learning architecture for visual change detection," In: Leal-Taixé L., Roth S. (eds) Computer Vision – ECCV 2018 Workshops. ECCV 2018. Lecture Notes in Computer Science, vol. 11130. Springer, Cham. https://doi.org/10.1007/978-3-030-11012-3_10

[8] Y. Wang, Q. Chen, Q. Zhu, L. Liu, C. Li, and D. Zheng, "A survey of mobile laser scanning applications and key techniques over urban areas," *Remote Sens.*, vol. 11, no. 13, pp. 1–20, 2019.

[9] W. Xiao, B. Vallet, K. Schindler, and N. Paparoditis, "Street-side vehicle detection, classification and change detection using mobile laser scanning data," *ISPRS J. Photogramm. Remote Sens.*, vol. 114, pp. 166–178, 2016, 10.1016/j.isprsjprs.2016.02.007.

[10] P. Xiao, X. Zhang, D. Wang, M. Yuan, X. Feng, and M. Kelly, "Change detection of built-up land: A framework of combining pixel-based detection and object-based recognition," *ISPRS J. Photogramm. Remote Sens.*, vol. 119, pp. 402–414, 2016, 10.1016/j.isprsjprs.2016.07.003.

[11] W. Xiao, B. Vallet, M. Brédif, and N. Paparoditis, "Street environment change detection from mobile laser scanning point clouds," *ISPRS J. Photogramm. Remote Sens.*, vol. 107, pp. 38–49, 2015, 10.1016/j.isprsjprs.2015.04.011.

[12] R. Qin and A. Gruen, "3D change detection at street level using mobile laser scanning point clouds and terrestrial images," *ISPRS J. Photogramm. Remote Sens.*, vol. 90, pp. 23–35, 2014, 10.1016/j.isprsjprs.2014.01.006.

[13] Y. Zhan, K. Fu, M. Yan, X. Sun, H. Wang, and X. Qiu, "Change detection based on deep siamese convolutional network for optical aerial images," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 10, pp. 1845–1849, Oct. 2017.

[14] E. Guo *et al.*, "Learning to measure change: Fully convolutional siamese metric networks for scene change detection," *CoRR*, vol. abs/1810.09111, 2018. [Online]. Available: http://arxiv.org/abs/1810.09111

[15] Z. J. Yew and G. H. Lee, "City-scale scene change detection using point clouds," in *Proc. Int. Conf. Robot. Automat.*, vol. abs/2103.14314, 2021. [online]. Available: https://arxiv.org/abs/2103.14314

[16] R. Qin, J. Tian, and P. Reinartz, "3D change detection-approaches and applications," *ISPRS J. Photogramm. Remote Sens.*, vol. 122, pp. 41–56, 2016, 10.1016/j.isprsjprs.2016.09.013.

[17] B. Gálai and C. Benedek, "Change detection in urban streets by a real time Lidar scanner and MLS reference data," in *Proc. Int. Conf. Image Anal. Recognit., Ser. Lecture Notes Comput. Sci.*, In: Karray F., Campilho A., Cheriet F. (eds) Image Analysis and Recognition. ICIAR 2017. Lecture Notesin Computer Science, vol. 10317. Springer, Cham. https://doi.org/10.1007/978-3-319-59876-5_24

[18] C. Benedek, "3D people surveillance on range data sequences of a rotating Lidar," *Pattern Recognit. Lett.*, vol. 50, pp. 149–158, 2014, 10.1016/j.patrec.2014.04.010.

[19] J. Bromley *et al.*, "Signature verification using a "siamese" time delay neural network," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 7, no. 4, pp. 669–688, 1993.

[20] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comp.-Ass. Interv.*, 2015, pp. 234–241.

[21] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2017–2025.

[22] A. Börcs, B. Nagy, and C. Benedek, "Fast 3-D urban object detection on streaming point clouds," In: L. Agapito, M. Bronstein, C. Rother (eds) Computer Vision - ECCV 2014 Workshops. ECCV 2014. Lecture Notes in ComputerScience, vol 8926. Springer, Cham. https://doi.org/10.1007/978-3-319-16181-5_48

[23] C. E. Metz, "Basic principles of ROC analysis," *Seminars Nucl. Med.*, vol. 8, no. 4, pp. 283–298, 1978.

[24] P. Luc, C. Couprie, S. Chintala, and J. Verbeek, "Semantic segmentation using adversarial networks," in *Proc. NIPS Workshops Adversarial Training*, Dec. 2016. [Online]. Available: https://hal.inria.fr/hal-01398049

[25] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1231–1237, Sep. 2013.

[26] J. Schauer and A. Nüchter, "Removing non-static objects from 3D laser scan data," *ISPRS J. Photogramm. Remote Sens.*, vol. 143, pp. 15–38, 2018, 10.1016/j.isprsjprs.2018.05.019.