Ray Tracing Assisted Radar Detection in 6G

Ilkka Moilanen, Timo Lintonen, Markku Kiviranta
VTT Technical Research Centre of Finland Ltd
Oulu, Finland

Pekka Sangi, Juha Pyhtilä, Pekka Pirinen, Markku Juntti
Centre for Wireless Communications, University of Oulu
Oulu, Finland

ABSTRACT. In this paper, we present a novel method that detects appearing targets and improves their detection probability in a known static environment. Ray tracing-based channel model is used to calculate the radio signal paths using a 3D model of the environment. This preliminary information and the measured radar image are delivered to the convolutional neural network (CNN) based autoencoder (AE). The output image provides an improved representation of appearing targets, as the redundancy coming from known static environment is modeled and cancelled from the radar image. The method does not need the traditional constant false alarm rate (CFAR) threshold setting, which is an advantage, especially in heterogenous environments. The functionality of the method is tested with radar measurements in a laboratory corridor. Our results show that environmental clutter decreases significantly in the radar image and new targets are more clearly distinguished.

Keywords—radar, ray tracing, autoencoder, JCAS, 6G

I. INTRODUCTION

Reliable and robust radar sensing is one enabler for the forthcoming sensing applications. The 6G envisions the design of joint communications and sensing (JCAS) or the integrated sensing and communications (ISAC). The ISAC/JCAS enables an explosive increase in the number of radar functionalities, and communication network can be viewed as a sensor. Network sensing typically refers to the detection of the presence or absence of objects, their shape, location, and/or speed of movement using radio signals transmitted and received by network elements [1]. Reliable environmental awareness is not achieved just by increasing the number of sensors. The angular resolution of the radar depends on the number of antennas and the range resolution is inversely proportional to the bandwidth. In the 5G, multiple-input and multiple-output (MIMO) antenna systems are already in use, and it also brings wider bandwidth with dense small cell deployments.

These enhancements increase the suitability of the ISAC/JCAS functionalities for various applications.

The minimization of clutter and reliable interpretation of the radar image takes on a significant role in many heterogenous environments. In conventional radar devices, constant false alarm rate (CFAR) algorithms are widely used to object detection. However, if the number of guard cells and training cells, as well as the probability of false alarm are set improperly, the target detection performance is severely degraded [2]. In the future, artificial intelligence (AI) or machine learning (ML) -based solutions will take on an even more important role to detect and isolate the object of interest from the environmental clutter. In [2], a convolutional neural network (CNN) -based autoencoder (AE) object detection was proposed to replace the CFAR algorithm in the MIMO frequency-modulated continuous wave (FMCW) radar systems.

In this paper, we combine data-driven ML approach to ray tracing (RT) based physical channel modelling. Ray tracing is a method which calculates the radio signal paths using 3D model of the environment. Channel modelling information is used together with a radar image to improve radar object detection accuracy.

To summarize our method, both ray tracing and radar information are delivered to the AE. The AE suppresses the reflections that have been modeled through ray tracing from the known static environment. After suppression, new events can be detected more accurately in the radar image. The functionality of the method is tested with radar measurements in the laboratory corridor. The 60 GHz FMCW MIMO radar [3] and the fast ray tracing simulator [4] are developed by VTT Technical Research Centre of Finland Ltd and the University of Oulu, respectively.

II. PROPOSED METHOD

We propose a novel method that obtains preliminary information from reflections through ray trace modeling to enhance the detectability of appearing targets in radar images. In more detail, the reflections from the known static environment that fall within the radar's field of view are modelled. This preliminary information is then used together with the radar image in the CNN-based AE to improve detection. Another benefit of the method is that the AI-based detection also eliminates the need for traditional CFAR algorithm, which is an advantage, especially in heterogenous environments where the threshold setting is challenging. At the top hierarchy level, the setup includes a radar

device, a ray tracer and a CNN based AE block. As presented in Fig. 1, the AE block receives range-angle (RA) map inputs from both the radar and ray tracer.

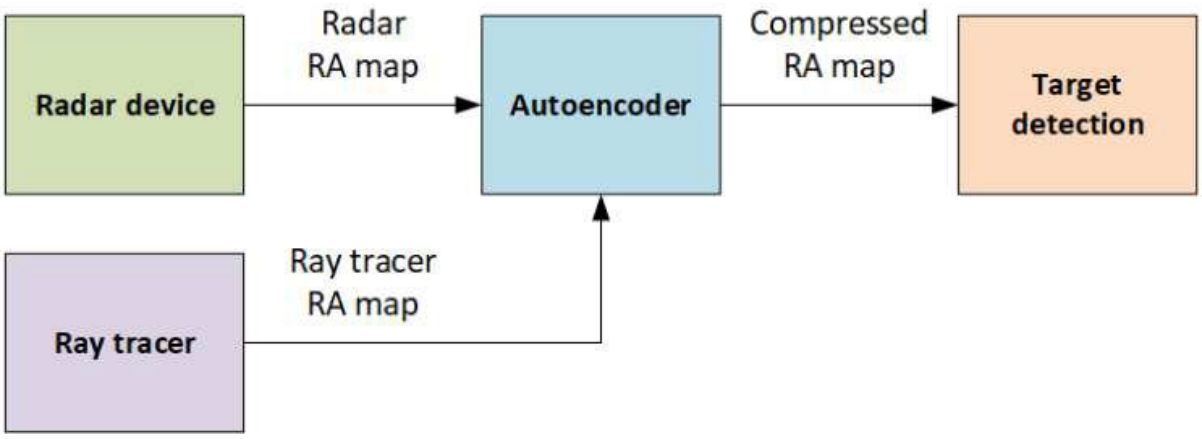

Fig. 1. Block diagram of the proposed method.

The RA maps illustrate the powers of the reflected signals at different distances at a certain angle as shown in Fig. 2. The range from 0.5 to 6.0 m is represented by 110 values on the y-axis. Similarly, the angles from −80 to +80 degrees are mapped by 180 values on the x-axis.

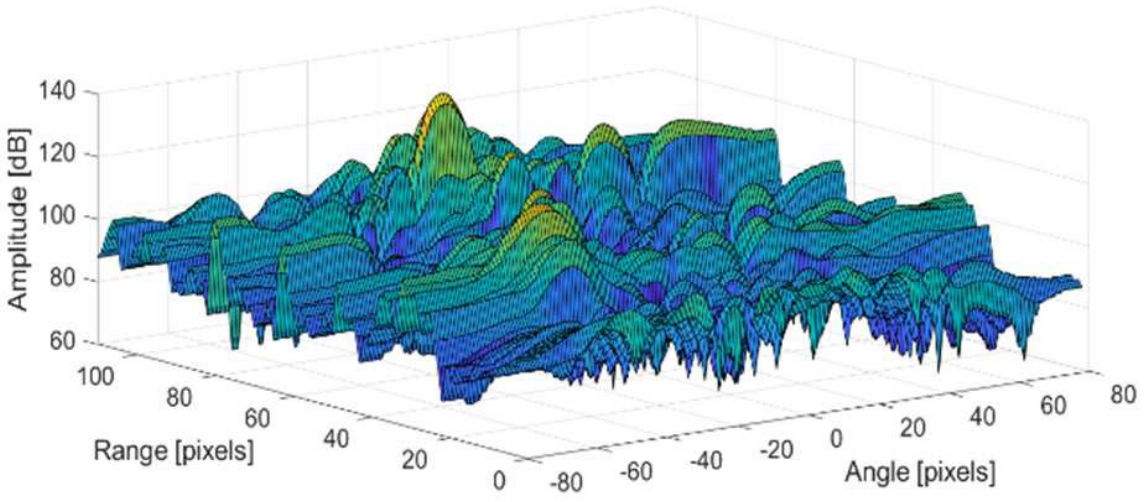

Fig. 2. Example of a RA map.

In general, the AE detector has a particular bottleneck property which can now be used for compressing the relevant information and ignoring noise and redundancy. For this purpose, we first assume that the bottleneck of the autoencoder compresses the information from images of 110×180 pixels into a 36-dimensional vector. Then the decoder decodes it back to an RA image. The process is then confirmed based on how much the inputs and the reconstruction differ from each other. The method is flexible.

The compressed RA image in Fig. 1 can only contain the reflections from the radar target. But if needed, the RT modelled reflections from the static environment can be reintegrated into final image after detecting an appearing target to include known static objects into same image. We can also define the mask generated by the ray tracer so that only the second or higher order reflections from the known static environment are removed from the final output image.

A. Radar Detection

In the future, ISAC/JCAS devices can be used to form raw radar images instead of traditional radar device. However, since the proposed method is independent of the selected radar, we are using VTT's in-house 60 GHz FMCW MIMO radar [3] of which scalable model is illustrated in Fig. 3. In our case, the device has 4 TX and 8 RX antenna elements which results in 32 virtual channels. The radar bandwidth is 3 GHz. As shown in Fig. 4, the radar transmits a frequency chirp whose frequency increases linearly during the transmission.

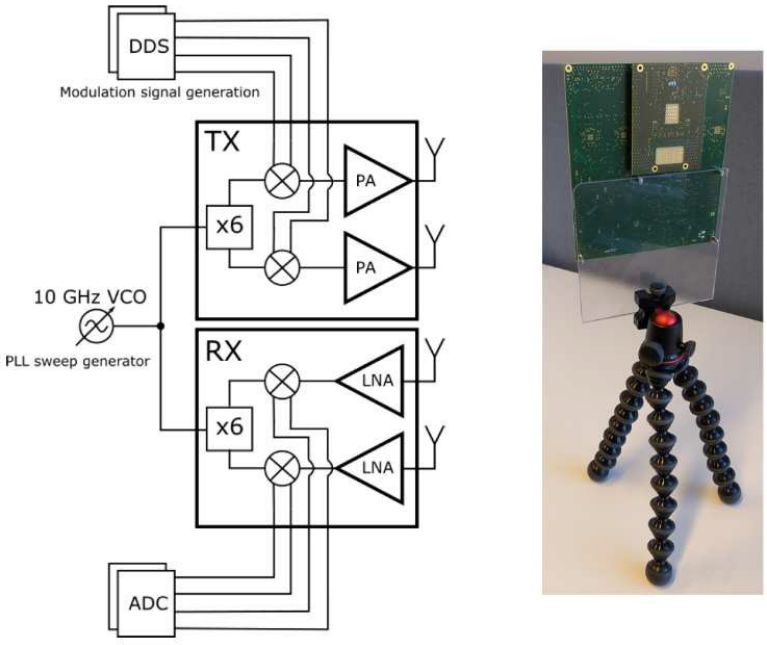

Fig. 3. Conceptual system model and radar hardware.

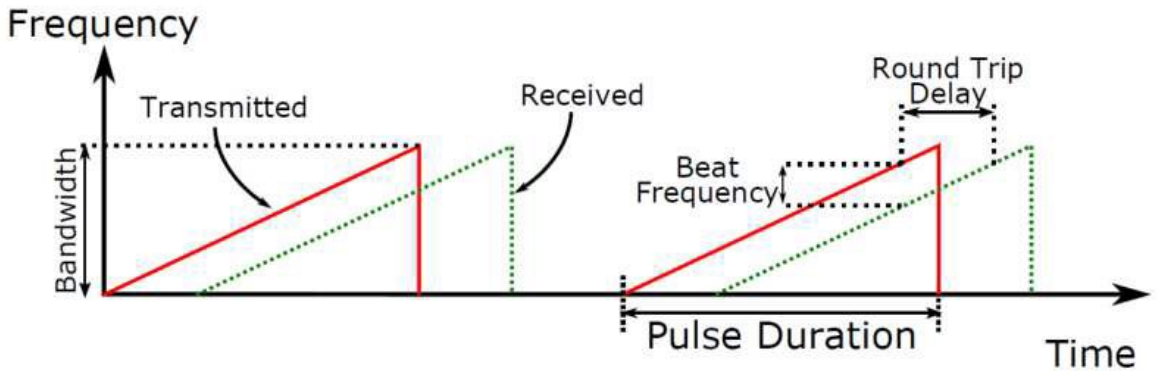

Fig. 4. FMCW chirp and principle.

The radar uses a frequency division-based MIMO solution, which introduces a slight frequency shift between the chirp sequences of the TX antennas. For example, TX1 and TX2 have the same frequency sweep, but these sweeps are 1.0 MHz apart. All the RX channels receive all the TX channels. The TX channels are separated in the digital backend. In range estimation, the differences in frequency between the transmitted and the received delayed signal are measured. In other words, the received signal is correlated in mixer with the transmit signal, which results in a signal called beat signal.

Then fast Fourier transform (FFT) is calculated over samples of the beat signal to obtain the spectrum, which has peaks corresponding to the targets at different ranges. However, it is important to keep in mind that the peak position in the studied TX channel includes the frequency shift related to frequency division, and this must be taken into account in the detection.

When all TX-RX combinations are extracted from the received signal, digital beamforming is used to build 2D image of the radar field of view. In practice, digital beamforming is done by using FFT. As a result, RA map of size 110×180 is achieved. With the abovementioned parameters, the range and angular resolutions are 5 cm and 3.5 degrees, respectively. The RA map can be viewed as an input for the autoencoder processing in Fig. 1

B. Ray Tracing

As presented in Fig. 5, the ray tracing simulations are implemented using a two-phase approach, where point-to-point electromagnetic (EM) path models are calculated first. This provides a basis for antenna simulations done in the second phase. The point-to-point propagation channel model computation is done with a tool described in [4]. Using a discretization point-based representation

of the geometry, visibilities between point pairs are determined as presented in [5]. Candidate routes from TX to RX points are then searched, using the NVIDIA OptiX ray trace engine [6]. Resulting paths are refined using a gradient descent-based technique and validated using OptiX. Finally, EM models are computed for valid paths [4]. Information about the angle of departure and arrival (AoD, AoA), propagation delay, intermediate points, and interaction types are stored for each path, in addition to EM features. In the second phase, results of point-to-point computation are utilized by simulating the antennas at TX and RX points. For radar, simulated beamforming with varying antenna steering produces RA images, as explained next.

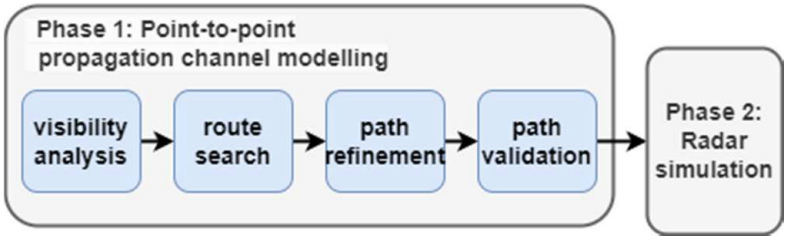

Fig. 5. Dataflow in ray tracing (RT) based channel model computation.

1) EM Modelling

The paths derived in the ray tracer are used to evaluate the EM field properties along the paths and at the receiver. We employ the geometric optics approximation for the paths, treating EM radiation propagating in a specific direction as a ray tube that does not interact with neighboring rays. [7]. The wave fronts of the ray tube and the electric and magnetic fields are orthogonal to the propagation direction. By considering the interaction point locations, as well as the geometries and other properties of the material, an exact output electric field vector at the terminating point of the ray can be calculated.

The output electric field vector is computed by evaluating a 2×2 polarimetric transformation matrix [8] using the Fresnel equations and the uniform theory of diffraction [9] at each interaction point along the path.

The TX/RX antenna is modelled using its specifications and the theoretical cavity model [10] for individual patches. The antenna model efficiently outputs a field vector consisting of information for both field amplitude and polarization. This enables the complete EM analysis of the field along a path, considering both reflections and diffractions. The field polarization for the RX antenna enables

the computation of the antenna's effective length, which is needed for the exact calculation of the received signal power. The TX/RX field patterns derived with the help of the cavity method are depicted in Fig. 6. These antenna patterns are compared to actual measurements and found to be reasonably accurate.

Differences are evident near the antenna plane denoted in Fig. 6 by ±90 degrees. The cavity model predicts a zero H-plane field at ±90 degrees, whereas full wave simulations and measurements give a non-zero field.

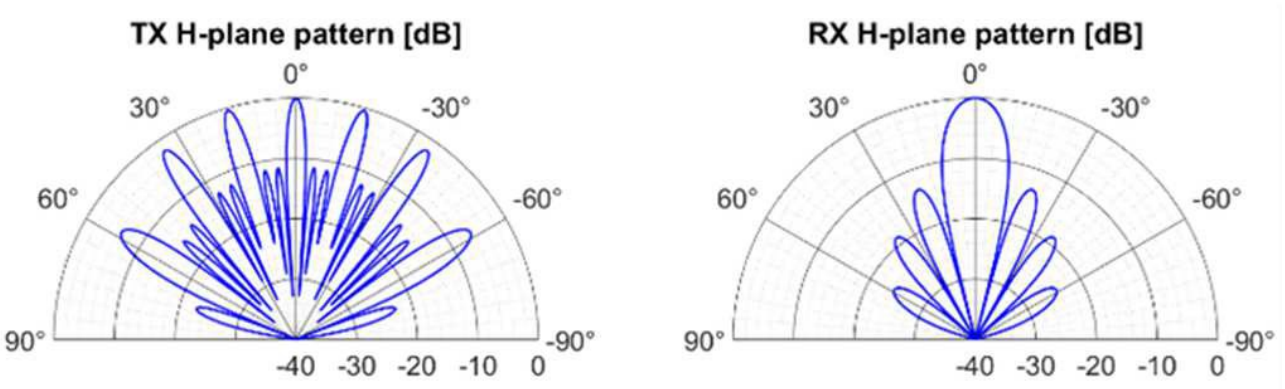

Fig. 6. The 60 GHz unsteered antenna patterns in the horizontal plane.

2) Range-Angle (RA) Image Production

Simulated RA images are obtained with the computational phases shown in Fig. 7. Simulation of radar beamforming is done for 180 angle points in the range from −80 to +80 degrees and as a result, an estimate of the power delay profile (PDP) for each simulated angle is obtained. In the signal processing phase, received signal powers are estimated for 110 distances in the range matching with the radar setup from 0.5 to 6.0 m in measurements. Time delays T associated with the multipath components are mapped to the difference of current received and transmitted frequencies of the FMCW radar. As a result, we produce a weighted sum of sinusoids and analyze this signal with FFT to obtain one column of the RA image. The interpretation of FFT bins as specific distances assumes single interaction along the paths. Then, the time delay T corresponds to the range

$$r = cT/2 \tag{1}$$

where c denotes the speed of light.

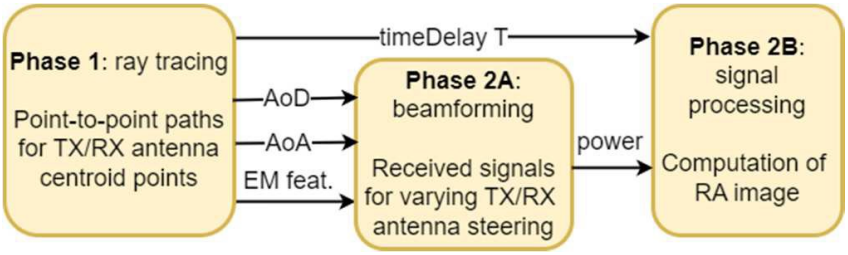

Fig. 7. RT simulation-based computation of radar RA image.

3) Simulation Results

The combination of radar measurements and ray tracing with the AE is tested in the laboratory premises. For this purpose, the geometry of the corridor located at the University of Oulu was modeled using the Blender tool [11]. The resulting GLTF file was converted to the triangle mesh-based format used by the ray tracer, using a tool that also detects and marks potential diffraction edges. The resulting model consisted of 2136 triangles and 101 edges. Discretization for the geometry was done with a fine 0.25 m resolution, which provided 16761 tiles and 1240 edge segments. Ray tracing to obtain point-to-point paths was conducted by limiting the number of interactions to 3 and the diffraction count to 2. Using some pruning for weak paths, the setup provided 3145 TX-RX paths, which were reduced to 2552 paths in antenna simulation. In Fig. 8, the simulation result is shown with an image of the corresponding empty corridor measurement. There are only a few paths involving one interaction, and a huge number of paths (due to the closed indoor environment) come from multiple interactions. Those paths contribute to the simulated RA image, which has a similar fan-like echo pattern as the measurements. Some differences are also visible. The wall on the right from the radar's viewpoint provides, e.g., a couple of echoes at about 1.8 and 2.7 m distances, which are not visible in the simulation.

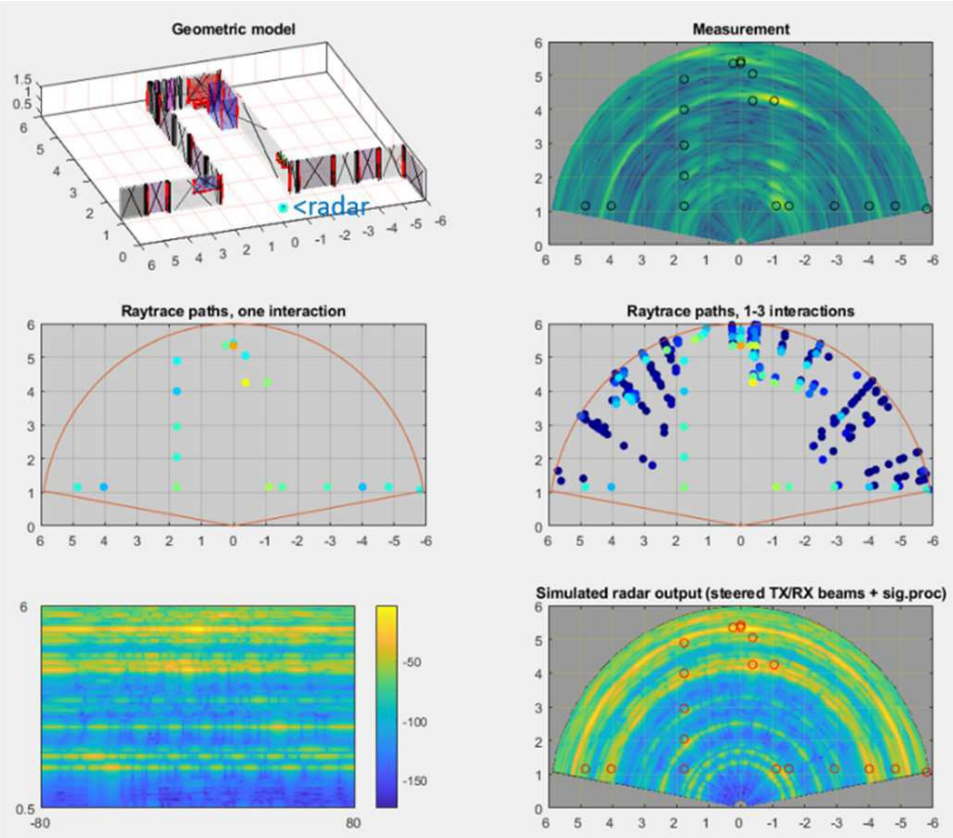

Fig. 8. Ray trace simulation result. Top: geometric model and measured RA image. Middle: ray tracing result with AoA and propagation delay thresholds. Bottom: simulation-based RA image with beamforming and signal processing emulations.

C. AI Processing

We use AI to automate the target detection in the RA maps. Our model aims to differentiate between an empty corridor and a detectable target. The goal is achieved through novelty detection [12] that separates novel observations from the expected data. We train our AE model using only RA maps from an empty corridor, and these RA maps constitute our expected data. This then means that any target entering the room should generate a novel observation. To further aid the learning of the model, we incorporate interpretation of the known static environment in the form of a RT mask.

1) Neural Networks

The AE is a powerful unsupervised neural-network model that can learn a compact representation of the observed data [13]. The AE acts as an identity-function estimator using bottleneck to compress the inputs [14]. As presented in Fig. 9 and Fig. 10, the AE is now composed of two neural networks, namely an encoder and a decoder. First, the AE encodes the RA map into a lower-dimensional vector representation.

Then, the AE reconstructs the RA map by decoding that vector. The weights of the AE are learned by minimizing a reconstruction error that measures how much the model outputs differ from the RA map inputs. As the figures show, the AE model is based on a DenseNet [15] backbone that is known for its densely connected CNN layers [16]. The dense connectivity improves the gradient flow throughout the network enabling easy training of very deep networks. In our AE model, we utilize dense blocks that consist of bottleneck layers of depth twelve with growth factor of four repeating three times. The down-sampling transitions consist of a convolutional layer with 3×3 kernel and an average pooling with 2×2 kernel and stride two. The up-sampling transition is transposed convolution with 3 ×3 kernel and stride two. Each convolutional layer is preceded by batch normalization and Rectified Linear Unit (ReLU) activation.

The encoder in Fig. 9 consists also of down-scaling convolutional layer with 5×5 kernel and stride two and two repetitions of dense block followed by down-sampling transition. The last average pooling results in a 36-dimensional vector. Experimentally, we found that 36-dimensional encoding space is a reasonable compromise between overfitting and underfitting models. The decoder in Fig. 10 starts with a fully connected layer that upscales the 36-dimensional vector into a size 27×45 image. This image is then expanded into a tensor of depth twelve using a convolutional layer with 3×3 kernel. This tensor is then fed into two repetitions of dense block followed by up-scaling transition. Finally, the size 110×180 output is recovered by a convolutional layer with 3×3 kernel.

In the first phase, we bring a RT mask into our AE model to enhance the spatial awareness. First, RT finds the shape of the corridor (i.e., distances to the walls and obstacles) for K angles $\{\alpha_1^{RT}, \dots, \alpha_K^{RT}\}$. The distance to the closest obstacle is interpolated for all angles $\alpha \in [-90°, 90°]$ by taking the maximum distance of the five closest angles α_k^{RT} with $k \in 1, \dots, K$. The locations of the obstacles are then encoded into a matrix **M** whose shape corresponds to the shape of the RA map matrix **X**. The elements in the matrix **M** are '1' for each bin in the RA map matrix **X** that is inside the corridor. The elements are '0' for each bin that is behind a wall or an obstacle. Then, the RT masked RA map is **M** $\odot$ **X** where the operator $\odot$ is the Hadamard product.

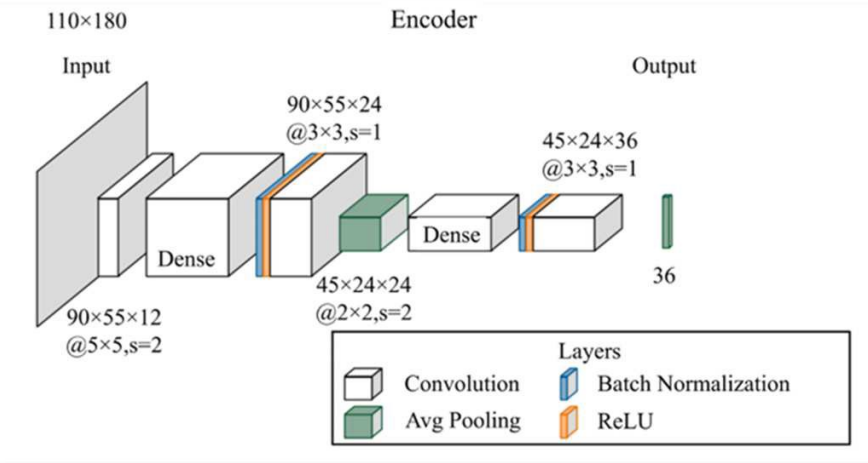

Fig. 9. CNN encoder.

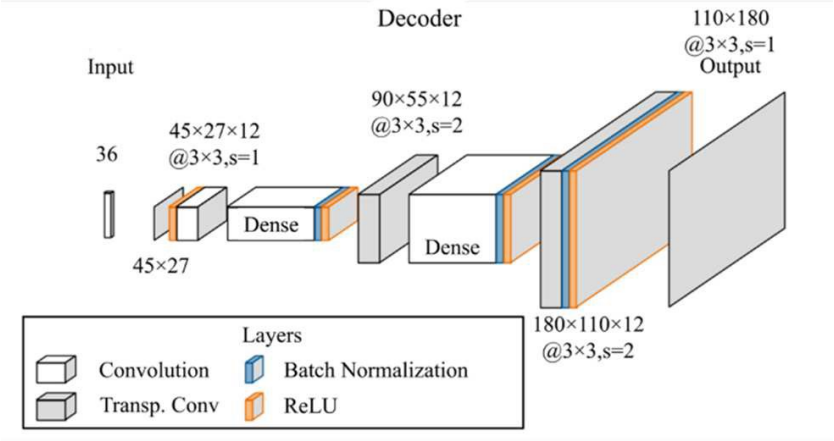

Fig. 10. CNN decoder.

In the second phase, beamforming and radar signal processing emulations are added as explained in the context of Fig. 7. Using this kind of RT based simulated RA map, we can reduce the environmental clutter from the radar resolution cells that depict the area inside the corridor. Let **S** be the simulated RA map matrix. The training data for our AE is the RT-enhanced RA map defined as $\mathbf{M} \odot (\mathbf{X} - \mathbf{S})$. Finally, we can train the AE to implicitly learn the RT-enhancement by using the enhanced RA maps $\mathbf{M} \odot (\mathbf{X} - \mathbf{S})$ as the target values in the reconstruction loss similarly to denoising [17].

2) Calculating Detections

The expected observations typically provide a lower mean squared error (MSE) [13] than novelties [12]. According to the three-sigma rule [18], the RA map **X** is a novel observation if it yields a reconstruction error

$$e_r > \hat{\mu} + 3\hat{\sigma} \tag{2}$$

where the variables $\hat{\mu}$ and $\hat{\sigma}$ represent the sample mean and standard deviation in the training set. Given a novelty that is generated by a target entering the empty corridor, we then aim to locate the target. This can be achieved by inspecting which bins in the RA map **X** are responsible for the rise in the reconstruction error. We do this by comparing the masked RA map to the AE output. Let us first denote encoder and decoder by functions $e(\cdot)$ and $d(\cdot)$. Then we can define an error matrix

$$\mathbf{E} = (\mathbf{M} \odot (\mathbf{X} - \mathbf{S}) - d(e(\mathbf{X})))^2 \quad (3)$$

where element e_{ij} is defined by indices *i* and *j* that correspond the range and angle bins, respectively. The error matrix **E** includes high-frequency noise that may lead to a faulty localization. To mitigate this, we leverage the information in the neighboring bins. In more detail, we assume that the error matrix **E** exhibits elevated values not only in the bin corresponding to the target location but also in its neighboring bins. To control for the noise, we may use the Gaussian blur [19]. Let us convolve the error matrix **E** repeating the convolution *l* times using a Gaussian kernel of size 5. The location of the target can then be found in the maximum bin in the blurred error matrix **E**. We found experimentally that *l* = 10 applications of Gaussian blur were enough to offset the noise in the error matrix **E**.

We trained our models on Ubuntu 22 with 8GB of RAM running Pytorch on Python. We used 8192 RA maps for training AE and another 2048 RA maps to estimate the sample mean and standard deviation of the reconstruction error. We minimized the reconstruction error using the Adam with the learning rate of 10^{-3} [20]. The limiting factor of the CNN AEs is the simultaneous requirement to load both the model and data into memory. The above-described model (see Fig. 9 and Fig. 10) was the largest trainable model when using a batch size of 32 RA maps. The reconstruction loss stopped improving after 10 epochs.

III. Validation of the Model

To prove the functional concept of the proposed method, we performed radar measurements in the laboratory corridor. As shown in the left-hand subfigure Fig. 11, we made the first measurements at an empty corridor having only a fixed safety box and medicine cabinet. Then, we added a radar reflector being a metal pyramid as an appearing target at different distances shown in the right-hand subfigure Fig. 11. The goal of the CNN based AE analysis is then to detect the pyramid and calculate its position relative to the radar. As presented in Section II, by utilizing the RA map generated from the ray tracer as training data for the AE, we can remove clutter from the known environment and enhance the visibility of appearing targets compared to the raw radar RA map. Now, the RA map of the radar amplitude signal is presented in the upper left subfigure in Fig. 12. Respectively, the bottom left subfigure of Fig. 12 illustrates the RA map from the ray tracer. For convenience, these reflections are interpreted in more detail in the upper right subfigure. These interpretations are based on manual inspection of the photograph of the corridor in Fig. 11. The ray-tracing

model automates this interpretation step enhancing spatial awareness. It also increases the accuracy of interpretation, as seen in the ability to detect the medicine cabinet. The cabinet is difficult to notice from the radar signal because it is so close to the wall. In the AE, the redundant information of the inputs is removed, and the result is presented in the bottom right subfigure of Fig. 12. The removal of redundant information is divided into two steps. First, all reflections coming from outside the corridor are removed by using the estimated shape information of the corridor, and then known reflections inside the corridor are cleaned up. The output data from the AE provide a much more precise representation of appearing targets since the noise coming from outside the corridor can be safely ignored. Radar resolution cells inside the corridor are also cleaned from the known clutter of the static environment. In the ideal case, only the reflector would be visible on the radar image after cleaning. In practice, some clutter will of course remain because the ray tracer cannot perfectly model the clutter visible in the radar image.

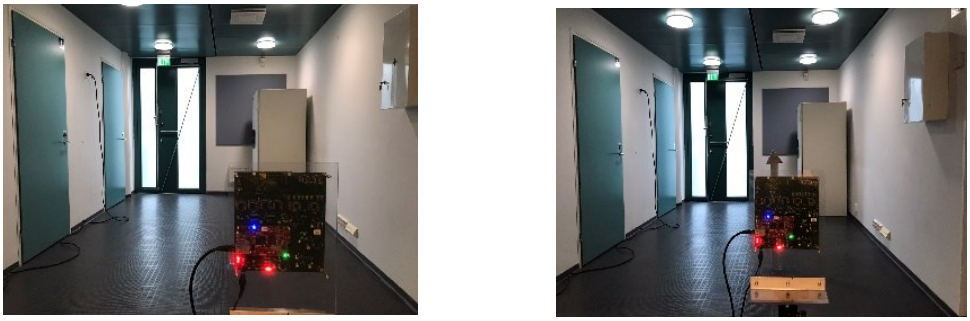

Fig. 11. Test environment with and without reflector.

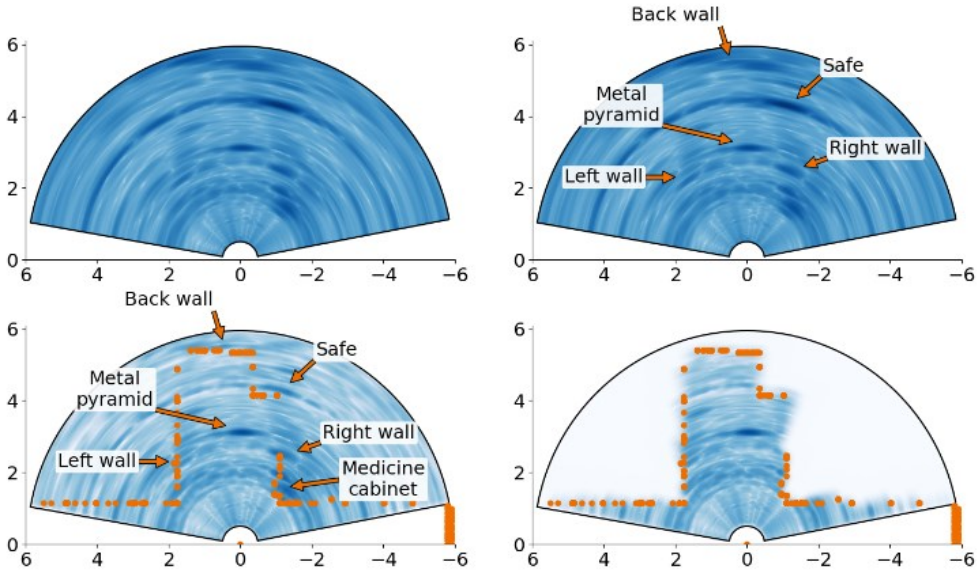

Fig. 12. The upper left panel shows radar signal, while the upper right panel displays manually interpretable shadows from a corridor photograph. The bottom left subfigure displays precise environmental information from the ray-tracing signal, while the bottom right subfigure illustrates the clutter-free output of the AE with ray tracing.

The more detailed output of the AE detection-model is illustrated in Fig. 13. In this example case, the metal pyramid reflector is positioned in front of the radar at 3025 mm distance (see right hand subfigure of Fig. 11). The output of the detection model conforms to the shape of the corridor. The most notable feature of the output is a clear spike at coordinates (0, 3). This is the position of the reflector, which can be automatically found from the output simply by finding the maximum of the output.

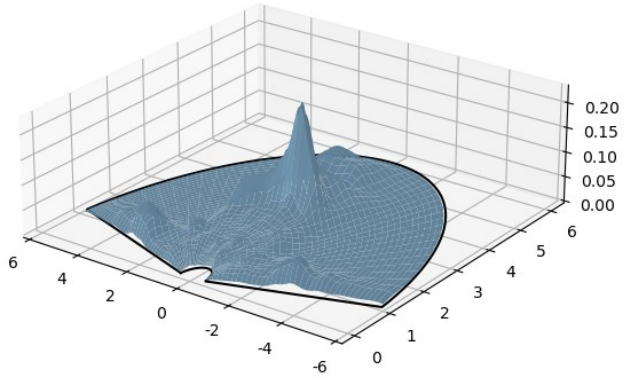

Fig. 13. The AE clearly detects the reflector at 3025 mm from the radar.

IV. Conclusions

We proposed an AI-based radar target detection method that uses ray trace modeled prior information about the environmental clutter to enhance the accuracy of detecting appearing targets. In the method, the CNN-based AE receives RA map inputs from both the radar and ray tracer and removes redundancy. The AE offers a significantly more accurate representation of objects since it disregards noise originating from outside the considered area, i.e., it can ignore areas behind static obstacles. It also removes modeled environmental clutter from the radar resolution cells that depict the area inside the area of interest. We validated the performance of the method in laboratory corridor using the 60 GHz radar hardware. We found that the method significantly reduces clutter caused by the known static environment. If necessary, the objects from the static environment can be added back into the final image after the detection of an appearing target. The method also allows to customize the mask generated by the ray tracer. For example, we can define a mask so that only the second-order reflections from the known static environment are removed from the final output image. In the future, the performance of the method will be evaluated in more complex clutter environments. Outdoor measurements will also be included in the testing campaign.

Acknowledgment

This work has been supported in part by the Business Finland under the projects AI4Green and 6GLearn, with the corresponding diary numbers 1052/31/2019, 1267/31/2019 and 8603/31/2022, 8738/31/2022. The work

of University of Oulu has also been supported in part by the Academy of Finland, 6G Flagship program under Grant 346208.

References

[1] T. Wild, V. Braun, and H. Viswanathan, “Joint design of communication and sensing for beyond 5G and 6G systems,” in IEEE Access, vol. 9, pp. 30845-30857, 2021.

[2] S. Kang, M. Jang, and S. Lee, “Autoencoder-based target detection in automotive MIMO FMCW radar system,” Sensors, vol. 22, July 2022.

[3] H. Forsten, T. Kiuru, M. Hirvonen, M. Varonen, and M. Kaynak, ”Scalable 60 GHz FMCW frequency-division multiplexing MIMO radar,“ IEEE Trans. Microw. Theory Tech., vol. 68, July 2020.

[4] N. Vaara, P. Sangi, J. Pyhtilä, M. Juntti, and J. Heikkilä, “A refined path generation pipeline for radio channel propagation modeling,” in Proc. EuCAP, 2023.

[5] J. S. Lu, E. M. Vitucci, V. Degli-Esposti, F. Fuschini, M. Barbiroli, J. A. Blaha, and H. L. Bertoni, “A discrete environment-driven GPU-based ray launching algorithm,” IEEE Trans. Antennas Propag., vol. 67, Feb. 2019.

[6] S. G. Parker, J. Bigler, A. Dietrich, H. Friedrich, J. Hoberock, D. Luebke, D. McAllister, M. McGuire, K. Morley, and A. Robison, “OptiX: A general purpose ray tracing engine,” ACM Trans. Graphics, vol. 29, July 2010.

[7] C. A. Balanis, Advanced Engineering Electromagnetics. 2nd ed, John Wiley and Sons, New York, USA, 2012.

[8] A. F. Molisch, H. Asplund, R. Heddergott, M. Steinbauer, and T. Zwick, “The COST259 directional channel model-part I: Overview and methodology,” IEEE Trans. Wireless Commun., vol. 5, Dec., 2006.

[9] R. G. Kouyoumjian and P. H. Pathak, “A uniform geometrical theory of diffraction for an edge in a perfectly conducting surface,” Proc. IEEE, Nov. 1974.

[10] C. A. Balanis, Antenna Theory: Analysis and Design, 4th ed. John Wiley and Sons, New Jersey, 2016.

[11] B. O. Community, Blender - a 3D modelling and rendering package, Blender Foundation. [Online]. Available: http://www.blender.org

[12] B. Ouafae, L. Oumaima, R. Mariam, and L. Abdelouahid, “Novelty detection review state of art and discussion of new innovations in ihe main spplication fomains,” in Proc. IRASET 2020.

[13] H. R. Kerner, D. F. Wellington, K. L. Wagstaff, J. F. Bell, C. Kwan, and H. Ben Amor, “Novelty detection for multispectral images with application to planetary exploration,” in Proc. AAAI 2019.

[14] Y. Dong and M. L. Seltzer, “Improved bottleneck features using pretrained deep neural networks,” in Proc. INTERSPEECH 2011.

[15] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in Proc. CVPR 2017.

[16] R. Pitale, H. Kale, S. Kshirsagar, and H. Rajput, “A schematic review on applications of deep learning and computer vision,” in Proc. ASIANCON, 2021.

[17] Y. Xin, H. Chen, and L. Xie, “Efficient noisy data transmission using denoising autoencoder,” in Proc. ICSPCC, 2021.

[18] F. Pukelsheim, “The three sigma rule,” The American Statistician, vol. 48, May, 1994.

[19] J. Kostková, J. Flusser, M. Lébl, and M Pedone, “Handling Gaussian blur without deconvolution, Pattern Recognition, vol. 103, 2020.

[20] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in Proc. ICLR, 2015.