InvRT: Solving Radar Inverse Problems with Transformers

Anonymous submission

Abstract

A wide variety of applications rely on the characterization of objects from radar observations. Existing approaches in this space primarily focus on inferring object type. In this work, we focus on inferring the input parameters of a physics-based radar simulator from its output, a temporal sequence of radar observations. Sequential radar observations inherently have a complex spatio-temporal structure that is difficult to capture with many standard vision-based deep learning architectures. We model such complex phenomena as a sequence to sequence prediction problem and use a transformer architecture, taking advantage of its ability to capture contextual temporal dependencies. We demonstrate that our method, Inverse Radar Transformer (InvRT), outperforms baseline approaches in predicting object properties, for both high and low observability settings. Furthermore, its errors are highly correlated with the level of object observability, highlighting its potential to learn the geometric limitations of radar sensing.

Introduction

The advantages of radar technology, such as the ability to search wide areas and to operate in all weather conditions, have led to its widespread use in object detection for autonomous cars (Bilik et al. 2019), robotics (Stetco et al. 2020), and biosensing (Diraco, Leone, and Siciliano 2017). It has additionally proven useful for weather and climate forecasting, such as detection of tornado debris (Cheong et al. 2017), ice crystals in clouds, (Myagkov et al. 2016), and rainfall estimation (Gorgucci et al. 2001), which will become increasingly critical in the future as society faces the effects of climate change. In such scenarios, object properties play a major role in making critical decisions. Additionally, solving nonlinear inverse physics problems with deep learning is the subject of ongoing research (Mason, Yonel, and Yazici 2017). Motivated by neural networks' capabilities as universal function approximators, these approaches involve estimating the inputs to physics-guided simulations, given the noisy outputs. In this work, we use transformers to solve the useful inverse problem of object property estimation from simulated sequential radar observations, using object shape as an example. Transformers are the current state-of-the-art deep learning architecture for processing sequential data (Vaswani et al. 2017). Although developed for natural language processing problems, recently they have significantly advanced the state-of-the-art in a variety of domains, including signal processing and computer vision (Dosovitskiy et al. 2020; Li et al. 2019; Guo et al. 2021).

Existing deep learning algorithms developed for radar object characterization primarily infer object class rather than object attributes (Geng et al. 2021). In addition, many such approaches only encode spatial information in individual radar signals by utilizing 1D convolutional neural networks (CNNs) (Lundén and Koivunen 2016a; Wan et al. 2019; Song et al. 2019; Lundén and Koivunen 2016b; Pan et al. 2022; Wan et al. 2020) or recurrent neural networks (RNNs) (Xu et al. 2019). Some of these approaches additionally apply attention mechanisms (Pan et al. 2022; Wan et al. 2020; Xu et al. 2019) and transformers (Zhang et al. 2021) to enhance spatial encodings, boosting algorithm performance. However, temporal structure is equally important as spatial structure, and all of these approaches only consider individual radar observations, rather than sequential observations over time.

We introduce the Inverse Radar Transformer (InvRT) which expands on prior work in the following ways: (1) InvRT directly infers radar object properties; (2) InvRT exploits the temporal structure of radar observation sequences over time, in addition to spatial structure in individual observations. We demonstrate that our approach can learn to infer object characteristics at a higher performance rate than the baseline approach, while staying consistent with inherent noise and observability constraints.

Methods

Our problem relates closely to the class of reconstruction-based inverse problems: given a set of input states y, a forward model $F(\cdot)$, and the measured output x that is subject to measurement noise ν:

$$x = F(y) + \nu, \quad \nu \sim \mathbf{N}(0, \Sigma),$$

the objective is to estimate y given noisy measurements x using a neural network architecture. The following describes our approach to data generation, the model architecture, and the training and evaluation methodology.

Data Simulation

We use a high-frequency radar simulation tool (Chance et al. 2022) that allows us to parameterize objects so that their

measurements are the summation of individual components. We define a set of individual components to create different objects consisting of circular flat plates, spherical hemispheres, and conical frustums (Figure 1). The Radar Cross Section (RCS) for each component is calculated using legacy software (Burt and Moore 1991). To simplify our architecture and analysis, our experiments consider collections of stacked frustums with flat plates on both ends and collections with one end being a flat plate and the other a hemisphere. Other component combinations can be handled with minor changes to the architecture.

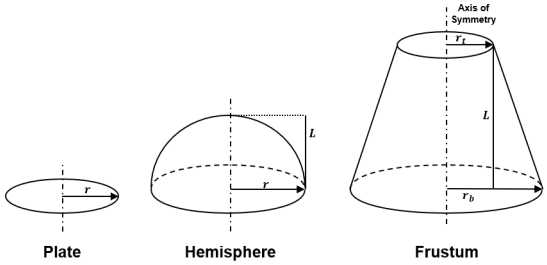

Figure 1: Geometry of the three main input shape components of the radar simulation.

The simulation input is a vector $y \in \mathcal{R}^{N_y}$ of parameterized circular roll-symmetric objects (Figure 2a). The simulation produces noisy radar observation, $x \in \mathcal{R}^{N_x}$. The observation is the normalized magnitude of the range-profile as described in Section III of (Chance et al. 2022) using waveform parameters provided in Figure 2c. The observation depends on the waveform, the dimensions of the components, and the angle between the line-of-sight vector and the object's axis of symmetry, which we call the aspect angle θ. An object is fully observed if the observations cover all aspect angles, $\theta \in [0, \pi]$ (an illustration is given in Figure 2d), otherwise it is partially observed. We generate observations such that each object is partially observed.

Model Architecture

The InvRT architecture consists of an embedding network, a conventional transformer backbone, and shape prediction heads (Figure 3). The embedding network uses a shallow 1-D CNN to learn a spatial embedding for each range profile. The CNN outputs are pooled across spatial dimensions to produce the final d-dimensional spatial embedding, where $d = 64$. A positional encoding is added to the spatial embedding.

The transformer architecture consists of 3 encoder and 3 decoder blocks, with the hidden dimension of all feed-forward layers and number of attention heads being 512 and 8, respectively. There are Q object component queries and 1 shape characteristic query (presence of a hemisphere) that are used in the attention layers of the transformer decoder. All queries are learned d-dimensional embeddings shared across all inputs. The decoder thus outputs Q component features and 1 shape characteristic feature for every input. Component features are locations on the object surface, a

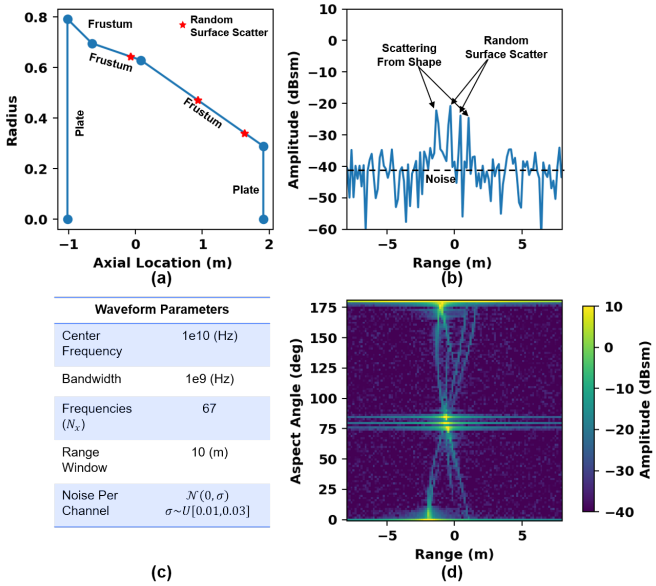

Figure 2: (a) Illustration of a random object cross-section. (b) The range profile for 140° aspect. (c) The waveform parameters for the simulation. (d) The range profiles stacked over all aspect angles.

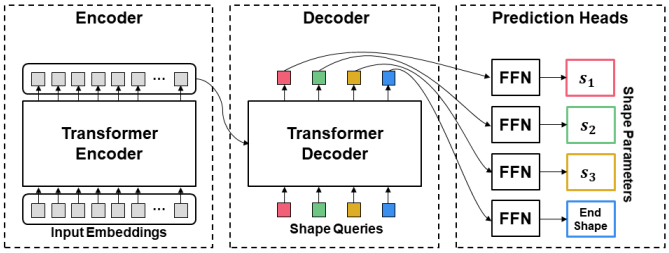

Figure 3: Overview of transformer architecture, adapted from (Carion et al. 2020). s_i corresponds to component i.

similar setup to (Carion et al. 2020). In this work, we set $Q = 20$. The component features are fed to parallel linear prediction heads: the classification head predicts a binary value for each component feature, and the regression head predicts (z, r) coordinates for each feature, where z is the location along the axis of symmetry and r is the radius.

During inference, InvRT only outputs coordinates for positively classified components. The shape characteristic feature is also fed to parallel linear prediction heads: the classification head classifies each end of the object as a hemisphere or flat plate, and the regression head predicts an axial z coordinate representing the tip of the hemisphere (the radius is zero at the tip of the hemisphere). During inference, InvRT only outputs this coordinate for end sections classified as a hemisphere. InvRT-Decoder has the same architecture as InvRT, except that it is a decoder-only transformer. All objects are post-processed to ensure geometric consistency (e.g. components representing flat plates are post-processed to have the same axial value, ensuring that

all flat plates are vertical).

Loss Function

The loss is a weighted sum of (a) the component classification loss, (b) the component regression loss, (c) the hemisphere detection loss, and (d) the hemisphere detection regression loss. We use the weights 1.0, 10.0, 0.5, and 0.5, respectively, chosen with careful hyperparameter optimization. (c) is the binary cross-entropy loss between labeled and predicted end shapes. (d) is the mean-squared loss between labeled and predicted hemisphere tip axial coordinates.

For a given component feature q, let the component classification logits be q_c and the regression output be $q_r = (z, r)$. The cost $C(p, q)$ between each ground truth component $p = (z, r)$ and q is defined as:

$$C(p, q) = -\log q_c + ||q_r - p||, \tag{1}$$

where $||\cdot||$ is the l_1 norm. Using this cost function, an optimal bipartite matching γ between true components and transformer outputs is computed with the Hungarian matching algorithm. We define (a) as:

$$\frac{1}{|Q|}(\sum_{q \in M} w_{pos} \log \sigma(q_c) + \sum_{q \notin M} \log\left(1 - \sigma(q_c)\right)), \tag{2}$$

where M is the set of matched transformer outputs and σ represents the sigmoid activation function. We set $w_{pos} = 3.0$. We define (b) as:

$$\frac{1}{|M|} \sum_{q \in M} ||q_r - \gamma(q)||^2, \tag{3}$$

where the matching function $\gamma(q)$ returns the ground-truth coordinates assigned to the component feature q, and $||\cdot||$ is the l_2 norm.

Model Training and Evaluation

All models were trained with 2 GPUs using 20,000 training objects and 1000 validation objects. They were trained with stochastic gradient descent (SGD) for 100 epochs with a constant learning rate of 0.001 and a batch size of 4. As each data sample comes from a noisy, inline simulation of an object, the amount of training data is the product of training objects and training epochs (2 million). Ten separate models were trained using differently seeded training and validation datasets. The data were augmented with random Gaussian noise. Varying the level of noise during training produced consistent trends, so we picked the range -40 to -30 dB for illustration.

Evaluation metrics consisted of: (1) intersection over union (IoU) between true and predicted object cross-sections and (2) hemisphere detection accuracy. Both metrics were evaluated over high and low observability settings. For (1), we define high observability as occurring when the range of observed aspect angles is greater than 90 degrees, and low observability when this is not the case. For (2), we define high observability as occurring when the median observed aspect angle is less than 90 degrees (thus centered on the front of the object), and low observability when this is not the case. Evaluation was run on the same holdout test dataset of 5000 testing objects, and the metrics were aggregated across all ten models using mean and standard deviation. To analyze performance sensitivity to signal degradation, all models were further evaluated on samples generated under signal degradation common in real-world radar applications, adding random Gaussian noise varying from -80 to -10 dB.

We compare the performance of InvRT and InvRT-Decoder against two baselines by replacing, where possible, the InvRT's encoder, decoder, or both with a LSTM recurrent network. Our goal is to evaluate whether the transformer architecture improves performance relative to other sequence networks. Note that we did not implement architectures with LSTM decoders and non-LSTM encoders, since LSTM decoders require hidden states as input and radar inputs are variable-length sequences. The baselines are trained in the same manner as InvRT and InvRT-Decoder.

Results

InvRT and InvRT-Decoder learned to predict object shape successfully in both high and low observability settings (Table 1) and outperformed both LSTM-based baselines. Since transformers have more capability to model long-range temporal dependencies than LSTMs, this indicates that the modeling of the full temporal sequence is critical for this task. We also note that although InvRT shows slightly better performance than InvRT-Decoder, the relatively similar performance of the two models demonstrates the power of decoder-only transformers, the architecture for many widely used algorithms (Brown et al. 2020). Last, even though we still observe high variability around the performance metrics, the InvRT and InvRT-Decoder models are able to reduce this variability in addition to improving average performance.

Furthermore, when the range of viewing aspects limits observability of the object, the InvRT trained on partial observations outputs higher uncertainty in inferring its shape. This effect is most pronounced when it comes to the hemisphere detection task (right side of Table 1), given that this task is expected to be the most susceptible to partial observability effects. However, it is interesting to observe that while InvRT methods performance degrades in a consistent manner with degraded observability, the LSTM baselines have comparative low performance in both low and high observability settings. This behaviour provides further indication that the InvRT models have improved performance due to their ability to better incorporate knowledge of the observed object geometry.

Figure 4 shows the performance of InvRT and InvRT-Decoder and both baselines at increasing levels of signal noise. The performance is measured as the cross-section IoU under both high and low object observability. InvRT and InvRT-Decoder consistently outperform the baseline across the different levels of modeled noise at both observability settings. This suggests that capturing the long-range temporal structure in radar signals with attention allows for better algorithm performance on degraded signals.

			Cross-Section IoU		Hemisphere Detection Accuracy (%)	
			Observability		Observability	
Method	Encoder	Decoder	High	Low	High	Low
InvRT	Transformer	Transformer	0.67 ± 0.11	0.63 ± 0.12	84.70 ± 36.00	64.56 ± 47.83
InvRT-Decoder	None	Transformer	0.61 ± 0.12	0.64 ± 0.11	80.86 ± 39.34	55.15 ± 49.73
LSTM-Transformer	LSTM	Transformer	0.45 ± 0.19	0.45 ± 0.20	55.86 ± 49.66	50.91 ± 49.99
LSTM	LSTM	LSTM	0.51 ± 0.14	0.51 ± 0.14	50.73 ± 49.99	49.88 ± 50.00

Table 1: Baseline Comparisons of Model Performance on Holdout Test Dataset

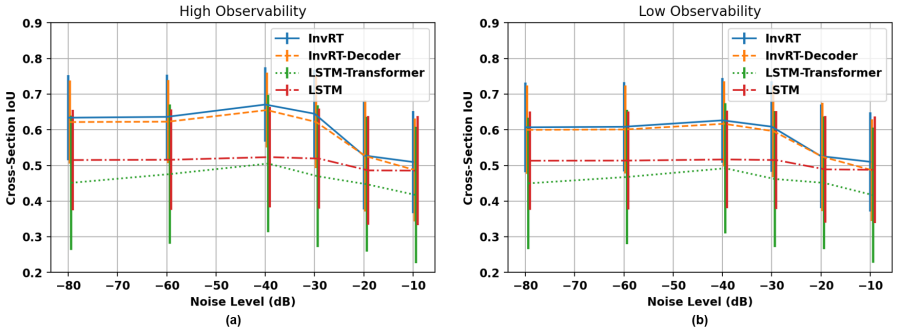

Figure 4: Algorithm sensitivity to signal noise in high (a) and low (b) observability settings.

Discussion and Conclusion

We have shown that transformers can be used to improve performance when solving the inverse problem of estimating object characteristics given radar observation data. Our work has several key advantages: (1) we model the problem as a sequence-to-sequence prediction problem and show that transformers can estimate object properties more effectively, when compared to other sequence-based architectures; (2) InvRT and InvRT-Decoder are better able to make predictions consistent with physical observability constraints than these baselines; and (3) they further outperform baselines on degraded signal inputs. (2) suggests that InvRT methods show potential for generative modeling of object property probability distributions given partial viewing histories. This is especially useful in high uncertainty cases when object observability is restricted and is a promising extension of our work. Furthermore, the algorithm can be adapted to predict other object attributes.

There are several limitations to our approach. Although the use of synthetic data allows our models to be trained on larger datasets and evaluated on some real-world sources of error (e.g. signal noise), it does not account for all sources, such as non-uniform time sampling. Additionally, the model is trained and validated on static radars and objects, but real-world equivalents often are moving. Thus, the next step for this work is to update the simulation and algorithm to better capture and generalize to real-world data.

References

Bilik, I.; Longman, O.; Villeval, S.; and Tabrikian, J. 2019. The Rise of Radar for Autonomous Vehicles: Signal Processing Solutions and Future Research Directions. *IEEE Sig. Proc. Magazine*, 36(5): 20–31.

Brown, T.; Mann, B.; Ryder, N.; et al. 2020. Language Models are Few-Shot Learners. In *Adv. in Neural Info. Proc. Sys.*, volume 33, 1877–1901. Curran Assoc., Inc.

Burt, E.; and Moore, T. 1991. High Frequency RCS Prediction Theory.

Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-End Object Detection with Transformers.

Chance, Z.; Kern, A.; Burch, A.; and Goodwin, J. 2022. Differentiable Point Scattering Models for Efficient Radar Target Characterization.

Cheong, B.; Bodine, D.; Fulton, C.; Torres, S.; Maruyama, T.; and Palmer, R. D. 2017. SimRadar: A Polarimetric Radar Time-Series Simulator for Tornadic Debris Studies. *IEEE Transactions on Geoscience and Remote Sensing*, 55(5): 2858–2870.

Diraco, G.; Leone, A.; and Siciliano, P. 2017. A Radar-Based Smart Sensor for Unobtrusive Elderly Monitoring in Ambient Assisted Living Applications. *Biosensors*, 7(4).

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2020. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *CoRR*, abs/2010.11929.

Geng, Z.; Yan, H.; Zhang, J.; and Zhu, D. 2021. Deep-Learning for Radar: A Survey. *IEEE Access*, 9: 141800–141818.

Gorgucci, E.; Scarchilli, G.; Chandrasekar, V.; and Bringi, V. N. 2001. Rainfall Estimation from Polarimetric Radar Measurements: Composite Algorithms Immune to Variability in Raindrop Shape–Size Relation. *Journal of Atmospheric and Oceanic Technology*, 18(11): 1773 – 1786.

Guo, M.; Cai, J.; Liu, Z.; Mu, T.; Martin, R.; and Hu, S. 2021. PCT: Point Cloud Transformer. *Comp. Visual Media*, 7: 187–199.

Li, S.; Jin, X.; Xuan, Y.; Zhou, X.; Chen, W.; Wang, Y.; and Yan, X. 2019. Enhancing the Locality and Breaking the Memory Bottleneck of Transformer on Time Series Forecasting. In *Adv. in Neural Info. Proc. Sys.*, volume 32. Curran Associates, Inc.

Lundén, J.; and Koivunen, V. 2016a. Deep learning for HRRP-based target recognition in multistatic radar systems. In *IEEE Radar Conf.*, 1–6.

Lundén, J.; and Koivunen, V. 2016b. Deep learning for HRRP-based target recognition in multistatic radar systems. In *IEEE Radar Conf.*, 1–6.

Mason, E.; Yonel, B.; and Yazici, B. 2017. Deep learning for radar. In *2017 IEEE Radar Conf.*, 1703–1708.

Myagkov, A.; Seifert, P.; Bauer-Pfundstein, M.; and Wandinger, U. 2016. Cloud radar with hybrid mode towards estimation of shape and orientation of ice crystals. *Atmospheric Measurement Techniques*, 9(2): 469–489.

Pan, M.; Liu, A.; Yu, Y.; Wang, P.; Li, J.; Liu, Y.; Lv, S.; and Zhu, H. 2022. Radar HRRP Target Recognition Model Based on a Stacked CNN–Bi-RNN With Attention Mechanism. *IEEE Trans. on Geoscience and Remote Sensing*, 60: 1–14.

Song, J.; Wang, Y.; Chen, W.; Li, Y.; and Wang, J. 2019. Radar HRRP recognition based on CNN. In *IET Int. Radar Conf.*

Stetco, C.; Ubezio, B.; Mühlbacher-Karrer, S.; and Zangl, H. 2020. Radar Sensors in Collaborative Robotics: Fast Simulation and Experimental Validation. In *2020 IEEE Int. Conf. on Robotics and Automation*, 10452–10458.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All you Need. In *Adv. in Neural Info. Proc. Sys.*, volume 30. Curran Associates, Inc.

Wan, J.; Chen, B.; Liu, Y.; Yuan, Y.; Liu, H.; and Jin, L. 2020. Recognizing the HRRP by Combining CNN and BiRNN With Attention Mechanism. *IEEE Access*, 8: 20828–20837.

Wan, J.; Chen, B.; Xu, B.; Liu, H.; and Jin, L. 2019. Convolutional neural networks for radar HRRP target recognition and rejection. *EURASIP J. on Adv. in Sig. Proc.*, 5.

Xu, B.; Chen, B.; Wan, J.; Liu, H.; and Jin, L. 2019. Target-Aware Recurrent Attentional Network for Radar HRRP Target Recognition. *Sig. Proc.*, 155: 268–280.

Zhang, L.; Chang, H.; Wang, Y.; Li, Y.; and Long, T. 2021. Polarimetric HRRP recognition based on feature-guided Transformer model. *Electronics Letters*, 57(18): 705–707.