IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 10, 2021, accepted March 30, 2021, date of publication April 12, 2021, date of current version April 21, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3072551

Predicting LiDAR Data From Sonar Images

NIELS BALEMANS[1,2], PETER HELLINCKX[1], AND JAN STECKEL[2,3]
[1]IDLab, Faculty of Applied Engineering, University of Antwerp—imec, 2000 Antwerp, Belgium
[2]CoSys-Lab, Faculty of Applied Engineering, University of Antwerp, 2000 Antwerp, Belgium
[3]Flanders Make Strategic Research Centre, 3920 Lommel, Belgium

Corresponding author: Niels Balemans (niels.balemans@uantwerpen.be)

This work was supported by the Flemish Government (AI Research Program).

ABSTRACT Sensors using ultrasonic sound have proven to provide accurate 3D perception in difficult environments where other modalities fail. Several industrial sectors need accurate and reliable sensing in these harsh conditions. The conventional LiDAR/camera approach in many state-of-the-art autonomous navigation methods is limited to environments with optimal sensing conditions for visual modalities. The use of other sensing modalities can thus improve reliability and usability and increase the application potential of autonomous agents. Ultrasonic measurements provide, compared to LiDAR, a much sparser representation of the environment, making a direct replacement of the LiDAR sensor difficult. In this work, we propose a method to predict LiDAR point cloud data from an in-air acoustic sonar sensor using a convolutional stacked autoencoder. This provides a robotic system with high-resolution measurements and allows for easier integration into existing systems to safely navigate environments where visual modalities become unreliable and less accurate. A video of our predictions is available at https://youtu.be/jlx1S-tslmo.

INDEX TERMS Machine learning, ultrasonic sensing, computer vision, inverse problems.

I. INTRODUCTION

Autonomous robotics have proven to be tremendously useful for many applications in several sectors, going from manufacturing [1] over predictive maintenance [2], [3] to security and surveillance [4], [5]. The navigation of autonomous robots is done by processing the measurements of sensors in order to understand and anticipate the environment. As the sensor signals are used to make navigational decisions, it is evident that the accuracy and reliability should be as high as possible for the safety of the robot and its surroundings. The perception and understanding of the environment can be seen as a fundamental and probabilistic signal processing and computer vision problem and has therefore been a popular research topic for many years. Specifically, for the autonomous navigation of robotic systems, countless methodologies and novel approaches can be found in order to improve sensor measurement quality and environment understanding [6], [7]. As stated in [5] the application potential of unmanned autonomous vehicles has brought tremendous interest and popularity to the field but the vision and perception problems remain, especially for vehicles operating in harsh and difficult sensing environments. The current state-of-the-art robotic systems in the academic and industrial world primarily use Time-of-Flight (ToF) sensors (e.g. LiDAR), cameras and radar sensors [8], [9]. These sensing modalities are proven robust and reliable sensors that perform well in specific environmental conditions. However, visual sensing modalities (ToF and cameras) will prove less reliable in difficult or harsh environments, for example, limited visibility due to dust or fog [10], [11]. The measurements produced by these sensors in non-optimal sensing environment should not be solely trusted for making navigation decisions.

In previous work, our research group (CoSys-Lab) has developed a novel 3D in-air sonar sensor (eRTIS) [12], which is capable of generating accurate 3D images of the environment using ultrasound. In this work, we propose a method for learning a transformation of sonar measurements into LiDAR point clouds. The goal of this transformation is to predict how a LiDAR sensor (point cloud) would perceive a certain environment based on sonar measurements from our eRTIS sensor. This prediction will be used to improve the usability of sonar data as this allows for the predicted data to be used in existing state-of-the-art methodologies designed for LiDAR point clouds.

In general, this work proposes a method for transforming sensing modalities into the representation domain of other modalities using deep learning. The goal is to determine whether this inverse problem of approximating the environment perception of one modality based on the measurements

The associate editor coordinating the review of this manuscript and approving it for publication was Seung-Hyun Kong.

of another, can be done reliably and accurately. This can essentially be understood as an approximation of measurements of one sensing modality based on the measurements of another. This prediction method can benefit robotic navigation and computer vision in general in several ways. The first advantage of this approach is that the algorithms designed for specific modalities can still be used, even when that modality is not available in certain situations, by *converting* the measurements from another.

A second opportunity is the possibility to incorporate the prediction results in fusion algorithms. The information of the prediction can affirm the measurements from the original sensor and thus increase the data reliability. The difficulty in sensor fusion is that the data needs to be spatially, temporally, and geometrically aligned to obtain usable results. This problem can be mitigated by first converting the data to the desired modality and fusing the approximation with the original sensing modality. This allows to easily create a multimodal sensor system that can deal with different sensing conditions. Sensor fusion is a commonly used method to improve the accuracy and reliability of the data extracted from sensor measurements [13], [14]. Several different approaches exist to mitigate the effects of unreliable sensors and improve the measurement accuracy (e.g., extended Kalman filters [15]). Recently, heterogeneous multimodal sensor fusion using deep neural networks can be considered as the important novel approach for autonomous navigation [6], [8], [10]. This sensor fusion method exploits the behaviours of different sensing modalities by combining them into one complementary sensor system. The benefit of this multimodal approach is that the fused measurements will be positively affected by each sensing modality, as each modality measures fundamentally distinct aspects of the environment. By designing the sensor systems for specific tasks, the multimodal fusion technique can be optimized for different environments. The goal of multimodal sensor fusion is to use a set of sensors that complement each other in different situations, one modality can perform worse than another, but the overall performance should be less affected. We argue that the eRTIS in-air sonar sensor [12] is an excellent addition to multimodal sensing systems that should be capable of accurately sensing in difficult and harsh environments. In bad visibility situations, the LiDAR and camera sensors will perform worse or fail, the eRTIS sensor is still able to perform in this situation. This complementary behaviour of sensors is exactly what is necessary to develop robust multimodal sensing systems. While the eRTIS sonar sensor can be used to obtain robust navigation behaviour and motion primitives [16], the use of the eRTIS in multimodal sensor systems is still to be explored. To facilitate this incorporation, we experiment with the conversion of 3D sonar measurements into LiDAR point cloud data. We see our eRTIS sensor as an addition to multimodal sensing systems rather than a replacement. The modality prediction approach presented in this work can be seen as a means to easily integrate into multimodal sensing systems.

The rest of this paper is structured as follows: in section II we argue why our approach to this sonar to LiDAR conversion is a suitable method based on similar research problems. Section III provides an in-depth discussion of the proposed approach and the steps taken to obtain the results that will be discussed in section IV. Finally, in section IV conclusions are made from the overall research and potential future research directions are introduced.

II. BACKGROUND

We start by introducing inverse problems to better argue why machine learning, and more specific convolutional neural networks (CNNs), are a suitable approach for the conversion of one sensing modality to another. Inverse problems are mathematical problems where to goal is to obtain *hidden* information based on *outside* or *indirect* measurements [17]. Creating a medical image from tomography data, reconstructing an image from a blurred one, and more recently upscaling image resolution (super-resolution) are among popular examples of inverse problems [18]. Most inverse problems belong to a special class of mathematical problems, namely the ill-posed problems, which means that there is likely not one unique solution or that the solution is highly unstable with respect to the input measurements. It is evident that no one algorithm or mathematical method exists to solve an inverse problem, each problem should be carefully analyzed in order to find a suitable approach. However, solving ill-posed inverse problems is already a rich and well-developed research field in which many problems have already been overcome and a vast amount of knowledge exists in order to better identify a fitting approach (e.g. [19]). An in-depth textbook discussion on ill-posed inverse problems can be found in [17] and [20].

In short, inverse problems arise when it is not possible to directly measure the required information, due to safety or other constraints. Indirect measurements will instead be used, along with a mathematical model describing the relationship between the required information and the indirect measurements to approximately reconstruct the desired data. Based on this description of inverse problems it can be easily understood that the modality conversion aimed in this work can be considered as an inverse problem. We aim to approximate point cloud data, representing the environment just as a LiDAR sensor would have perceived that environment, based on sonar measurements in situations where the LiDAR sensor is less reliable. In other words, we intend to obtain LiDAR measurements without directly measuring with a LiDAR sensor. By analyzing sonar and LiDAR data, in figure 1, one can quickly conclude that sonar measurements are sparse representations of the LiDAR measurements due to the specular nature of ultrasonic reflections. Mathematically speaking, to make this approximation we need the relationship between the LiDAR and sonar measurements. Many inverse problems can be formulated as solving the equation:

$$y = H(x) + e \quad \text{with} \quad y \in Y, \quad x \in X \tag{1}$$

where y represents the measured data, which in this case are the sonar measurements, x represents the data we wish to

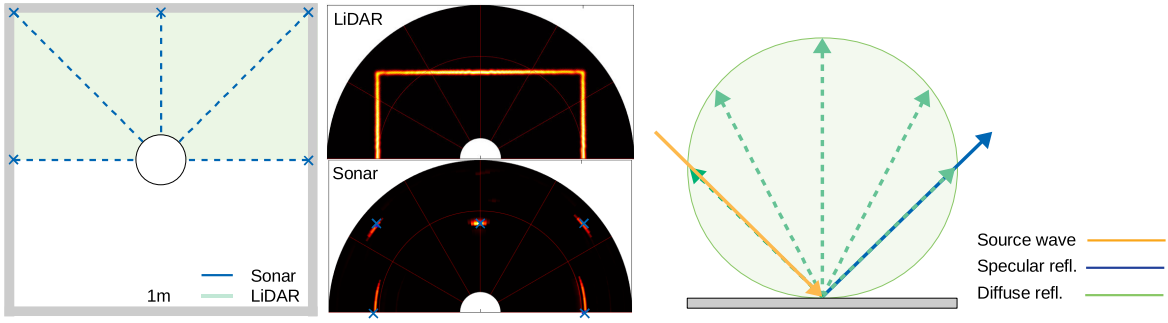

FIGURE 1. Overview of the differences between sonar and LiDAR measurements. The measurements of this simple environment demonstrate the differences between the ultrasonic and visual sensing modalities. The difference can be explained by the length of the used wavelength and the resulting reflection mechanism on (most) objects. The short wavelength of the LiDAR sensor will result in diffuse reflection, providing a denser representation of the environment, while the long wavelength of the sonar sensor will result in specular reflections and therefore a sparser environment representation.

reconstruct (LiDAR) and e represents a measurement error. The forward operator $H : X \rightarrow Y$ describes the relationship between a LiDAR data point and a sonar data point in the absence of measurement errors and noise. In theory, the inverse of this relationship H^{-1} can then be applied on the sonar measurements to approximately *reconstruct* the LiDAR point cloud. In [17] and [20], knowledge-driven approaches can be found for deriving the forward operator and a probability model for the measuring error in order to find a stable solution to the inverse problem. However, more recent research has argued for the use of data-driven approaches, and more specifically CNNs. The work in [21] and [22] provide a summary of recent studies that had great success and even improved state-of-the-art by using data-driven learning-based methods. Both works also provide a discussion on the contrasts between traditional approaches and the more recent data-driven methods.

The benefit of using a data-driven learning approach is that with a vast amount of available data the trained model can become very robust against input noise. It is also worth noting that CNNs, are forward non-iterative models that can be fully implemented in hardware, which can be important when the model is used for the control of vehicles and execution time is important. The disadvantage of using CNNs to solve ill-posed inverse problems is that large amounts of data are needed during the training phase to obtain robust results. This limits the use of the approach as measuring lots of data can sometimes be too difficult or expensive. In this research, however, we experiment with a way to overcome this issue, as a simulation is used to generate the training data. Further, we propose a deep convolution encoding-decoding neural framework in order to approximate this inverse relationship. Section III describes our approach to predicting LiDAR data based on sonar measurements in extensive detail.

In order to understand the difficulty of this conversion, and why it is impractical to use traditional methods, one needs to understand the inherent differences between the LiDAR and sonar sensors. It is more straightforward to understand the differences of the measurements when comparing the used wavelength of both sensors. The sonar sensor uses a relatively long wavelength as the used sensing modality is ultrasound. The LiDAR sensor, however, uses a visual modality (light), the wavelength can thus be considered as short. Both sensing systems rely on the reflections of the sent waves to reach back to the sensor in order to estimate the travelled distance.

Because of the long wavelength used by the sonar sensor, most surfaces will be seen as smooth surfaces, which results in specular reflections and means that the angle of the reflected wave is equal to the angle of the incident wave. This makes the received reflections significantly less, resulting in a sparse environment representation. The short wavelength of the LiDAR sensor enables diffuse reflections, as the wave is reflected on the (microscopically) small edges and imperfections on the surfaces. With a diffuse reflection, the wave is reflected in practically all directions, resulting in a more dense measurement. Figure 1 provides a visual representation of this phenomena and depicts the resulting differences in the measurements of both sensors. In nature animals such as dolphins and bats use ultrasonic echolocation effortlessly to obtain useful information from the environment. In many situations, the use of sound as a sensing modality can have many advantages over vision. However, it is still a difficult and complex task for a computer to use the ultrasonic measurements usefully. By using modality conversion, the measurements can be interpreted in another manner to increase the usability of the data tremendously. The following two subsections each highlight similar research in the conversion of sensor data to increase performance of navigation algorithms and usability of the data. Both studies also employ a data-driven learning method towards solving the inverse problem. The proposed methods also served as an inspiration to the approach we have taken in this work. Lastly, we also believe these works emphasize the use of CNNs as a suitable approach to solve these types of problems.

A. BATVISION

The work presented in [23] also supports our claim that the use of ultrasonic sensors can be extremely useful in situations where other sensors would not perform. They train a CNN to transform a binaural sound signal into a visual representation of the scene. The input of the network is obtained by ultrasonic measurements generated by one microphone, sending a

frequency-modulated chirp, and two microphones recording the echoes. The output of the network represents a depth map and a greyscale representation of the scene. To train the model a large dataset was used containing real-world synchronized echoes labeled with RGB images and depth maps. It is worth emphasizing that the model is trained using real-world measurements, instead of using a simulated approach.

In contrast to the sonar sensor used in this work, we used a novel sonar sensor created within our research group which provides a wide field-of-view with accurate 3D localization using 32 microphones [12], [24].

B. EVENT-TO-VIDEO

In [25] a method for reconstructing video from an event data stream is proposed. The reasoning for this conversion comes from the idea that the performance of specialised algorithms designed for video data streams, is challenging to be achieved by algorithms designed for event data. This work shows that by transforming the event camera data stream using a trained CNN to a video stream the same algorithms can still be used and outperform the approaches directly applied to the event data. The resulting trained network even has benefits over using normal camera data. In contrast with a conventional camera, which captures each frame at a fixed rate, an event camera captures the brightness changes of each pixel asynchronously. This means that with the use of the trained network high temporal resolution and dynamic range with no motion blur can be achieved.

The video reconstruction is performed by a recurrent encoder-decoder neural network. It is interesting to note that while the network was trained using simulated data, the performance of the conversion on real-world data is still cleaner and more detailed than other similar research.

III. LIDAR POINT CLOUD PREDICTION

In the discussion above we argued using a deep neural network as an appropriate approach for the prediction of LiDAR point clouds based on the measurements of a sonar sensor. We interpreted the problem as an inverse imagining problem where the goal is to reconstruct LiDAR data using the measurements of the eRTIS sonar sensor developed by our research group [12]. We use the sonar data at the input of our network, the measurements are processed into energyscape images which similarly represent the environment as the LiDAR data. In these energyscape images, the measured ultrasonic reflections are represented with respect to their range and direction angle to the sensor. A sonar energyscape image, as depicted in figure 2, can be interpreted as a non-uniform, sparsely, seemingly random sampled version of the LiDAR measurements. The goal of our CNN will thus be to intelligently interpolate between the sonar data points. The CNN will thus learn a world model that enables the reconstruction of LiDAR data based on the sparse sonar measurements. We provide multiple sonar energyscape images as the input of the network, these images are used to incorporate time and provide more information about the environment. The use of multiple energyscapes gives the network the ability to differentiate walls from small point circle reflectors as large surfaces are not guaranteed to have many reflectors. Only edges and the echo at the normal of the surface and the sensor are measured. The energyscapes are obtained by a beamforming technique on the echoes measured by the eRTIS sensor, for more information about the sonar sensor and the

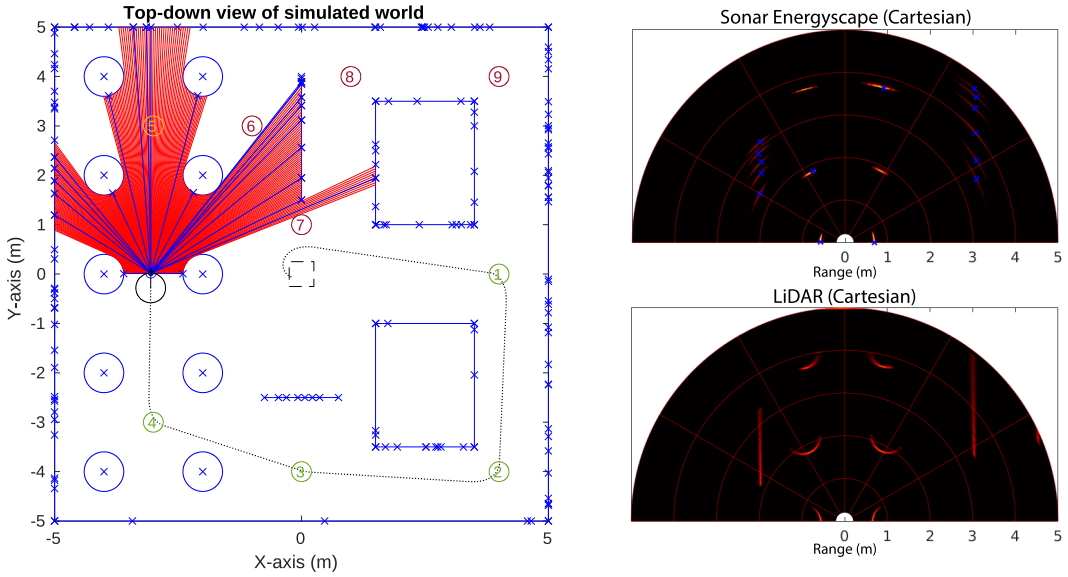

FIGURE 2. Example of a simulated scenario (left). The red lines represent the view of the LiDAR sensor, the blue lines depict the ultrasonic reflections that the sonar sensor receives. The blue crosses in the simulated environment are the ultrasonic reflectors, i.e. the points of an object the sonar sensor will measure. Each time step the sonar and LiDAR measurements are calculated and random noise is added to better approximate real-world data. The top measurement (right) represents the sonar energyscape of the current position of the robot. The bottom image represents the corresponding LiDAR measurement. This figure showcases the difference between LiDAR and sonar representations. The sonar energyscape is a non-uniform sparsely sampled version of the LiDAR measurement.

processing to obtain the energyscape images the reader is redirected to [12] or [24].

A. NETWORK ARCHITECTURE AND DESIGN

We opted for a stacked autoencoder network, mainly inspired by the UNet architecture proposed in [26]. In this encoder-decoder network, the sonar measurements are first encoded into a latent space. After further processing, the latent space representation is then decoded into the LiDAR point cloud representation. A high-level overview of the network can be found in figure 3. The input consists of the K most recent sonar energyscape images (with K = 1, 3 or 5 in our experiments), each sampled with equal time Δt between each measurement. We found that better results can be achieved by using multiple sonar images, this way the information of previous measurements can be propagated through time to obtain more reliable predictions at each step. The use of multiple sonar images also helps to deal with the limitations of ultrasonic sound as a sensing modality (e.g. the specular nature of ultrasonic reflections). The reflections measured by the sonar sensor are not dense enough to determine the complete shape of an object, the use of multiple measurements mitigates this limitation. Comparable results could also be obtained with the use of a recurrent neural network (RNN), which provides mechanisms for information to be kept within the network and propagated each forward pass. The training of such an RNN is however much more difficult as the memory of the network needs to be trained as well. This means that previous measurements are not guaranteed to be memorised and used in future forward passes, making it more difficult to converge the network to useful results.

The three encoding layers are followed by a residual layer, this block is inspired by the work in [27] and aims at increasing the learning capacity of the network, as well make it easier to train the deep convolutional neural network. The resulting output data is then decoded by the transposed convolution layers to obtain the predicted LiDAR point cloud. The residual block can therefore be interpreted as a mapping between the sonar latent space and LiDAR latent space.

Throughout the whole network, we use 3D convolutional layers, the kernel height differs in each layer, but the depth and width are fixed at K and 5 respectively. The fixed depth effectively means that we are using the K most recent sonar measurements for each prediction. A rectangular shape, instead of a conventional cubic shape, is used as kernel. The height changes between layers to compensate for the difference in resolution between the range and azimuth axis. Furthermore, we added a stride to the first and last layer to use a downsampled version of the input in the core of the network. Border effect problems are dealt with by the appropriate amount of padding, a detailed overview of all parameters is provided in table 1. We found the best results could be achieved by increasing the number of convolutional filters towards the center of the network.

TABLE 1. Network architecture details (K = 5).

Layer	Kernel	Stride	Padding	Filters
conv_1	(5, 13, 5)	(1, 5, 2)	(2, 6, 2)	20
conv_2	(5, 15, 5)	(1, 1, 1)	(2, 7, 2)	40
conv_3	(5, 15, 5)	(1, 1, 1)	(2, 7, 2)	80
residual_1	(5, 25, 5)	(1, 1, 1)	(2, 12, 2)	80
transp_conv_1	(5, 15, 5)	(1, 1, 1)	(2, 7, 2)	80
transp_conv_2	(5, 15, 5)	(1, 1, 1)	(2, 7, 2)	40
transp_conv_3	(5, 13, 5)	(1, 5, 2)	(4, 6, 2)	20

B. TRAINING AND DATASET GENERATION

We created a simulated and real-world measurements dataset to train the network. The real measurements dataset was mainly used to prove our approach works on real sonar energyscape images, while the simulated dataset was primarily used to experiment and tweak our model and hyperparameters. Training on simulated data has the advantage of easily creating a large number of labelled samples, as the model should be trained on many samples in order to achieve reliable and robust predictions. Another advantage of having a simulation model is being able to quickly extend and test upon this in future research. After training on simulated data, we tried to fine-tune the model on our real measurements dataset, the results of this are discussed in section IV.

1) SIMULATED DATASET

We created a robot simulation in MATLAB to easily generate large amounts of labelled data. This generation is done by traversing a simulated robot through an environment and calculating for each time step the corresponding sonar and LiDAR measurements. The sonar sensor is simulated as described in [16], for the LiDAR simulation we use a ray-tracing algorithm. We simulate the essential properties of both sensors as close to reality as possible, without making the computation too complex. To obtain a more robust model, we added random noise to the measurements. Simulating noise also helps to generate more realistic data, which can be useful to extend upon in future research. We assume the

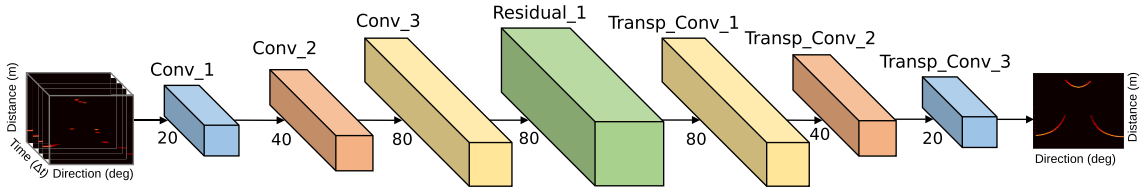

FIGURE 3. Overview of the proposed network architecture. The input of the network is composed of the K most recent sonar energyscape images. The convolutional and deconvolutional layers are each followed by an activation layer, specifically ReLU. The residual block consists of batch normalized convolutional layers and an activation layer, the goal of this residual structure is to increase the learning capacity of the network and ease the training of the deep convolutional neural network [27].

noise on physical measured data to be uniformly distributed for both the sonar and LiDAR measurements. This way we can simply add random values to the measurements, sampled from a uniform distribution. The validity of the simulation has been shown in previous research by the implementation of a closed-loop controller. This controller has been successfully deployed on a physical platform without tweaking the algorithm [16]. Figure 2 shows an example simulated environment and the corresponding simulated measurements. The simulator supports random world generation as well as load environments from SVG images to easily generate large datasets. Currently, only circle and wall objects are supported, these are arguably enough as most shapes can be approximated using these objects. We created approximately 12500 labelled samples using twelve different simulated scenarios. These scenarios differ for example in object size, rotation, or number of ultrasonic reflections.

2) REAL DATASET

For the creation of the real-world dataset, we mounted our eRTIS sonar sensor and a Hokuyo (UST-20LX) on a Turtlebot 2i and created measurements in an indoor environment. The measurements are synced using a synchronised clock for both sensors. We created approximately 5000 labelled samples, spread over hallways and office / cluttered rooms. Before training, the data was pre-processed to match angular and range resolution as well as minimum and maximum ranges. We used measurements between 0.2 to 5 m with an angular resolution of one degree, this is mainly determined by the sonar sensor.

The input of the network consists of five consecutive sonar energyscape images. For the corresponding ground truth LiDAR data, we used the data that was measured with the third sonar image. This introduces a time delay on the predictions of the model, but is necessary for more reliable results, as objects further away become less accurate.

During training, we used held-out validation with approximately ten percent of the data for which we ensured the validation samples came from scenarios that are not represented in the training data. To evaluate the generality of our model we also verified with k-fold cross-validation, the results of this are discussed in the results section. We used PyTorch [28] for the implementation and training of the network. We used a batch size of 12 and optimized the network with an Adam optimizer, as described in [29], with the initial learning rate set to 1e−4.

We used mean squared error loss (MSE), averaged over all output pixels (containing the LiDAR data). The MSE loss function was used as this provided the best experimental results. This network consists of over eleven million trainable parameters, to keep training time and computational resources within respectable bounds we did not use per-pixel loss. Lastly, we used the structural similarity index (SSIM), described in [30] as a performance metric to track learning progress and provide an early stopping mechanism when no or limited progress has been made over several epochs.

IV. RESULTS AND DISCUSSION

To verify the generality of our method, and to prove the trained model has learned a useful relationship between the sonar measurements and the LiDAR label we validated our model with k-fold cross-validation. We tested the model on samples that were never used during training and calculated similar performance metrics as during training. For these new samples, we calculated an average prediction accuracy using the MSE between the actual and the predicted LiDAR data. The training on simulated and real measurements results will first be discussed separately before a final discussion and conclusions are drawn at the end of this section.

Comparing the performance of our model to state-of-the-art models aimed at solving a similar problem is difficult, as to our knowledge no similar research results can be found transforming sonar measurements into LiDAR point cloud predictions. The work presented in [23] and [25] has a similar goal, but a direct comparison cannot be made as the used modalities are different.

A. TRAINING ON SIMULATED DATA

The cross-validation training progress and validation losses (MSE) for our network trained on simulated data are plotted in figure 5. We used k-fold cross-validation with k equal to eight and trained each fold for a maximum of 15 epochs (with approximately 12500 samples). In this plot, the mean and standard deviation of all folds is presented as well as the validation loss curves. What we can infer from this data is that our approach is able to generalise, and once convergence has been reached the model achieves high accuracy. At the start of training, the model performs very poorly, and predictions are not consistent (large std.) after approximately ten epochs the model performs equally for all validation samples (small std.). Figure 4 shows several model predictions with their respective ground truth LiDAR image. By empirically analysing the results in simulation, we can state that our trained model achieves very high visual accuracy. The delay introduced by the network becomes clear when comparing the predicted data with the latest LiDAR measurements. This effect becomes a dominant problem when the robot is moving at high speeds, however, high-speed vehicles are not the use case for our sonar sensor. This is because the speed of the sensor is dependent on the speed of sound, which limits the sensor from high-speed measurements. Another aspect of ultrasonic sound as a sensing modality that influences the accuracy of the predictions is the number of reflections received for an object. In LiDAR data objects can be easily identified and borders of the objects are easily detected. Our trained model guesses the locations where an object starts and ends based on the intensities and location of the ultrasonic reflections. It is clear, from the results in the simulator, that the predictions are heavily influenced by the number of reflections received from an object. In figure 4 images A and B showcase this effect. In 4-A a corridor can be seen with many reflectors on the walls, these reflectors can be interpreted as edges or details on the

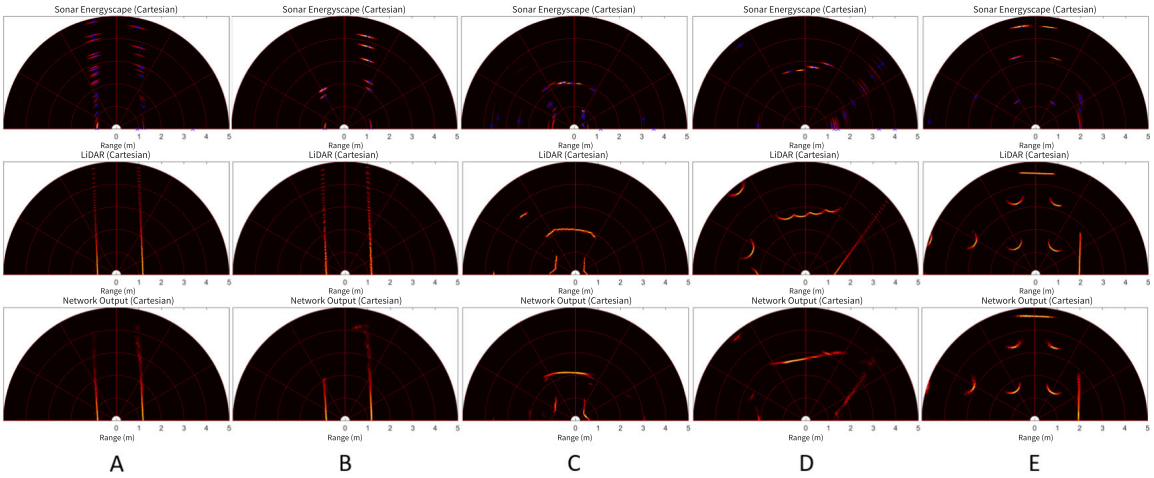

FIGURE 4. Prediction results of the trained model in the MATLAB simulator. Each column represents a scene for which (the most recent) sonar image is displayed at the top, the middle plot shows the ground truth LiDAR data, and our prediction results are presented in the bottom plot. A) Corridor with many reflectors. B) Corridor with a limited number of reflectors. C) Shows the model prediction for a more complicated object. D) Showcases the loss of detail in the prediction, the detail increases when the sensor gets closer to the object. E) Shows a predicted result that matches very closely with the ground truth and can thus be used for navigation.

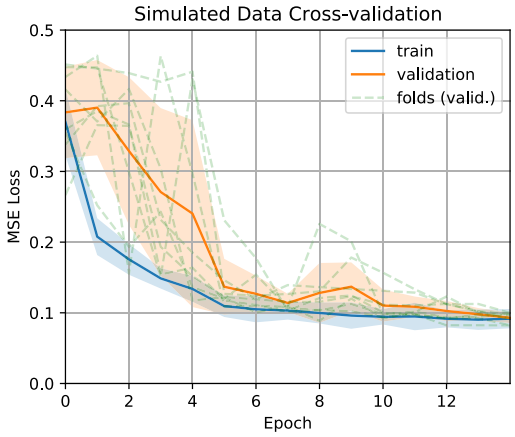

FIGURE 5. K-fold cross-validation (K = 8) training results with simulated data. The model was retrained eight times by splitting the dataset into eight data-folds and using a different fold as validation data each time. We present the mean and standard deviation over all training folds and plotted the validation loss curves for every fold. This plot shows that the model generalises once training convergence has been reached.

wall that reflect the ultrasonic sound wave. Figure 4-B shows the same corridor with the number of reflectors reduced, the model becomes less certain of the wall object and does not interpolate over the full object any more.

In order to numerically validate the performance of our approach, we simulated complex environments and calculated for each time step the mean square error (MSE). We tested several different scenarios: a limited amount of ultrasonic reflectors (only specular reflections), a random amount of reflectors and a random amount of reflectors with added noise to the input. To better understand the impact of the number of sonar energyscapes at the input, we performed these tests for three different networks, each with a different number of sonar images at the input (1, 3 and 5). Table 2 shows the average MSE over several runs for each case, the standard deviation is also calculated as a stability measure.

These results show that the amount of ultrasonic reflectors has a tremendous impact on the accuracy of the prediction results. The predictions are however still relatively accurate and stable. Based on this simulation we lose approximately 10% of accuracy in this worst-case scenario with very limited reflections. As can be expected, the amount of input noise also has a significant impact on the performance of the network. To enable future possibilities we verified that through retraining or fine-tuning the network with an adapted dataset the predictions can still be improved. The result of our fine-tuning is also presented in table 2. Finally, by comparing the different input sizes, we conclude that the model performs better when more sonar energyscape images are presented at the input. Figure 7 shows the performance of each tested input size in the simulated environment. We see that larger inputs perform better and more robust, especially in corners and situations whit a limited number of reflectors. However, the performance of the model with only one sonar energyscape at the input is also surprisingly accurate and robust. For future usage of the model, a trade-off should be made for the number of input images and the necessary output characteristics (e.g. time consistency, robustness, etc.)

TABLE 2. Analysis of LiDAR predictions for a complex simulated environment.

	Input Size	Minimal Refl.		Many Refl.		Noisy Input	
		Mean (μ)	Std. (σ)	Mean (μ)	Std. (σ)	Mean (μ)	Std. (σ)
MSE	1 sample	0.332096	0.087096	0.221591	0.090386	0.452140	0.059057
	3 samples	0.326139	0.090352	0.212811	0.079313	0.448643	0.058173
	5 samples	0.316683	0.100296	0.202580	0.086304	0.420531	0.063021
MSE	Retrain (5)	0.305678	0.096573	0.191520	0.093840	0.385094	0.114044

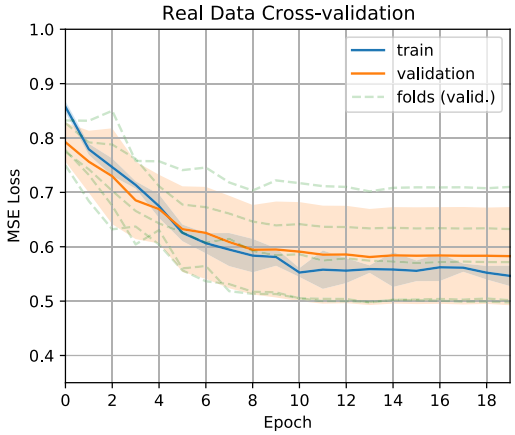

FIGURE 6. K-fold cross-validation (K = 5) training results with real-world data. The model was retrained five times by splitting the dataset into five data-folds and using a different fold as validation data each time. We present the mean and standard deviation over all training folds and plotted the validation loss curves for every fold. Compared to the simulated cross-validation results this model trained overall more stable. On this plot, the model looks generalised over the complete training session. However, samples are more alike than in our simulated dataset, as our real measurements were only created in an office environment.

with respect to increased calculation time a larger input entails.

To conclude, the overall performance of the sonar to LiDAR transformation in simulation is certainly suitable for making navigation decisions. As we have proven in the past, this simulation can be used to develop control algorithms that when applied on real-world vehicles still perform [16]. We will explore this further and work towards a navigation stack implementation using the proposed model in simulation.

B. TRAINING ON REAL DATA

Verification of our approach on real sonar measurements is important as the goal of this model is to improve the sensing and navigation capabilities of an autonomous agents in harsh sensing environments. We tried two methods to obtain a trained model for real ultrasonic acoustic images. First, we experimented with fine-tuning the simulated model on our real dataset. This experiment can be seen as a transfer learning step of the model trained on simulated data with real measurements. This would allow for training on the most training samples as we can then train on both simulated and real measurements to obtain a more robust and general model. However, due to some characteristics of ultrasound our simulation does not fully calculate, this fine-tuning step does not work optimally. Figure 8 presents a real sonar measurement that was used for the prediction of LiDAR data with our model trained on simulated data. It is clear the model has not learned that multiple or secondary acoustic reflections exist due to the lack of this characteristic in our simulated data. Therefore, we conclude that optimising the simulated model in a fine-tuning step for real-world measurements is not a viable option.

In order for the model to learn an accurate and robust transformation from real sonar images to real LiDAR point cloud data, we fully retrained the network on our real measurements dataset. The cross-validation loss curves are presented in figure 6. The mean and standard deviation over all data-folds are again plotted as well as the validation loss curves for each fold. At first glance, this figure also shows that the model generalises and achieves high accuracy when training convergence is reached. However, our real-world dataset only contains data from inside and office environments. It is evident that the samples are consequently more alike the samples created in simulation, where different scenarios are easily created. In figure 9 several real sonar measurements are presented with their respective LiDAR ground truth and model prediction. One can see that the trained network has learned a useful relationship between the sonar and LiDAR data as the model achieves a similar level of accuracy as the simulation. Our approach still performs, despite the different data characteristics compared to simulated data. The conclusions from training on simulated data also still hold, for example, the number of ultrasonic reflectors in the environment is still an important factor for the final quality and accuracy of the final result (e.g. figure 9-B).

We conclude that the use of CNNs to deal with inverse problems can be a suitable approach when enough labelled

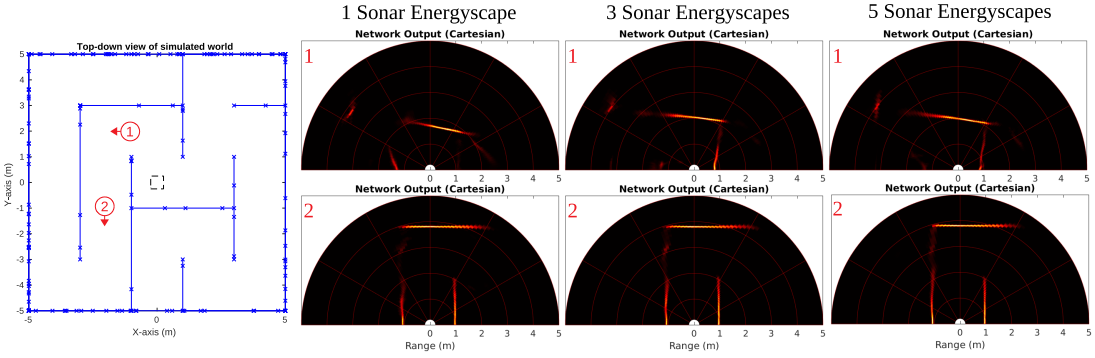

FIGURE 7. Influence of the number of input samples (energyscapes). Predicted LiDAR measurements on two locations in the simulated environment. The top row shows the impact of the number of energyscapes used on the result in a corner. The bottom row shows the results for a corridor with a limited amount of ultrasonic reflectors. Acceptable results can be obtained with one input sample, but the performance of the network is generally more stable and reliable when the number of input samples is increased.

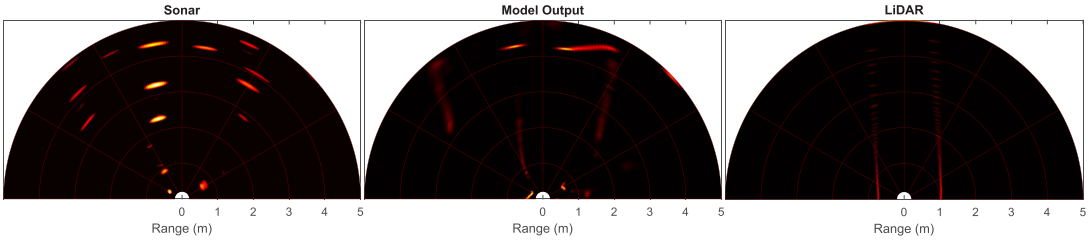

FIGURE 8. Prediction result of the model trained exclusively on simulated data when used to transform real sonar measurements. As our simulation does not simulate all characteristics of ultrasound the model cannot be directly applied to real data. It is evident that the model is not able to deal with characteristics that it has not seen during training. The sonar demonstrates (left) multiple or secondary reflections, the model is not able to deal with this phenomenon and has therefore not a correct result (middle) compared to the LiDAR ground truth (right).

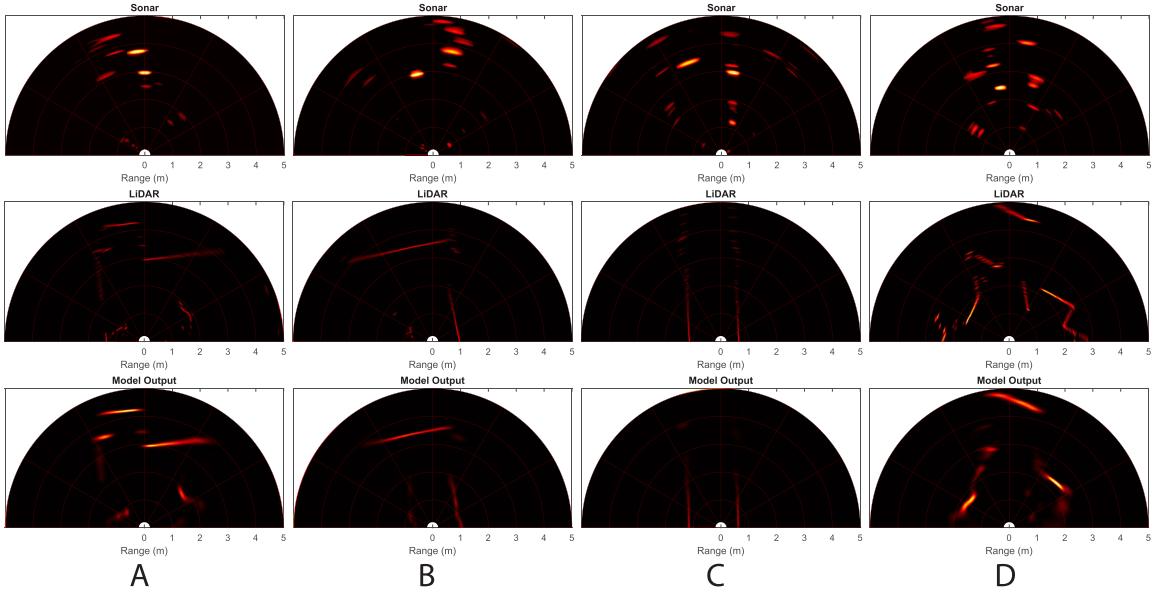

FIGURE 9. Prediction results of the trained model on our real sonar-LiDAR dataset. Each column presents a different scene for which (the most recent) sonar image is displayed at the top, the middle plot shows the ground truth LiDAR data and our prediction results is presented in the bottom plot. A) A complex indoor office environment, this result shows that the model can achieve very high accuracy. B) An office environment with less clutter than in A and thus fewer ultrasonic reflectors. C) A corridor environment, this result clearly demonstrate the multiple reflections received by the sonar sensor and the model being able to deal with this phenomena. D) Heavily cluttered scene, with lots of acoustic reflectors.

data can be obtained or generated. We cannot expect the approximation to contain as many details as the high-resolution LiDAR data would contain. The prediction will be a smoother, less accurate image of the scene, this can be observed in the results in figure 4 or 9. The main idea is that the closest obstacles are represented in the LiDAR prediction. As discussed above, our approach has limitations because of the ultrasonic sensing modality. Therefore the use case for this method is to complement existing sensing systems and increase performance in difficult and harsh sensing situations. This trained model functions as a proof-of-concept with which we defined a methodology that can be extended in future research. This work functions as a first step in our research on resource and context-aware robotic control. In future work, we will use and validate this model for LiDAR designed methodologies and experiment with adaptive robot control to use the best modality in a given situation. Finally, taking a closer look at the relationship the network has learned can be a fascinating research direction for further optimizations.

REFERENCES

[1] A. Gilchrist, "Introducing industry 4.0. in: Industry 4.0," in *Ind. 4.0*. Berkeley, CA, USA: Apress, 2016, pp. 195–215. [Online]. Available: http://link.springer.com/10.1007/978-1-4842-2047-4

[2] A. Schenck, W. Daems, and J. Steckel, "Airleak SLAM: Detection of pressurized air leaks using passive ultrasonic sensors," in *Proc. IEEE Sensors*, Oct. 2019, pp. 1–4. [Online]. Available: https://ieeexplore.ieee.org/document/8956631/

[3] C. Kyrkou, S. Timotheou, P. Kolios, T. Theocharides, and C. Panayiotou, "Drones: Augmenting our quality of life," *IEEE Potentials*, vol. 38, no. 1, pp. 30–36, Jan. 2019.

[4] D. Yanguas-Rojas, G. A. Cardona, J. Ramirez-Rugeles, and E. Mojica-Nava, "Victims search, identification, and evacuation with heterogeneous robot networks for search and rescue," in *Proc. IEEE 3rd Colombian Conf. Autom. Control (CCAC)*, Oct. 2017, pp. 1–6.

[5] M. Milford, S. Anthony, and W. Scheirer, "Self-driving vehicles: Key technical challenges and progress off the road," *IEEE Potentials*, vol. 39, no. 1, pp. 37–45, Jan. 2020.

[6] N. Patel, A. Choromanska, P. Krishnamurthy, and F. Khorrami, "Sensor modality fusion with CNNs for UGV autonomous driving in indoor environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 1531–1536. [Online]. Available: http://ieeexplore.ieee.org/document/8205958/

[7] C. Wong, E. Yang, X.-T. Yan, and D. Gu, "Adaptive and intelligent navigation of autonomous planetary rovers—A survey," in *Proc. NASA/ESA Conf. Adapt. Hardw. Syst. (AHS)*, Jul. 2017, pp. 237–244.

[8] V. De Silva, J. Roche, and A. Kondoz, "Robust fusion of LiDAR and wide-angle camera data for autonomous mobile robots," *Sensors*, vol. 18, no. 8, p. 2730, Aug. 2018. [Online]. Available: http://www.mdpi.com/1424-8220/18/8/2730
[9] D. Balemans, S. Vanneste, J. de Hoog, S. Mercelis, and P. Hellinckx, "LiDAR and camera sensor fusion for 2D and 3D object detection," in *Advances on P2P, Parallel, Grid, Cloud and Internet Computing* (Lecture Notes in Networks and Systems), vol. 96. Springer, Nov. 2020, pp. 798–807. [Online]. Available: http://link.springer.com/10.1007/978-3-030-33509-0_75, doi: 10.1007/978-3-030-33509-0_75.
[10] M. Bijelic, T. Gruber, F. Mannan, F. Kraus, W. Ritter, K. Dietmayer, and F. Heide, "Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Feb. 2019, pp. 11682–11692. [Online]. Available: http://arxiv.org/abs/1902.08913
[11] C. Papachristos, S. Khattak, F. Mascarich, and K. Alexis, "Autonomous navigation and mapping in underground mines using aerial robots," in *Proc. IEEE Aerosp. Conf.*, Mar. 2019, pp. 1–8. [Online]. Available: https://ieeexplore.ieee.org/document/8741532/
[12] R. Kerstens, D. Laurijssen, and J. Steckel, "ERTIS: A fully embedded real time 3D imaging sonar sensor for robotic applications," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 1438–1443. [Online]. Available: https://ieeexplore.ieee.org/document/8794419/
[13] J. Llinas and D. Hall, "An introduction to multi-sensor data fusion," in *Proc. IEEE Int. Symp. Circuits Syst.*, Dec. 1998, vol. 6, no. 1, pp. 537–540. [Online]. Available: http://ieeexplore.ieee.org/document/705329/
[14] R. C. Luo, C.-C. Yih, and K. Lan Su, "Multisensor fusion and integration: Approaches, applications, and future research directions," *IEEE Sensors J.*, vol. 2, no. 2, pp. 107–119, Apr. 2002. [Online]. Available: http://ieeexplore.ieee.org/document/1000251/
[15] L. Jetto, S. Longhi, S. Longhi, and G. Venturini, "Development and experimental validation of an adaptive extended Kalman filter for the localization of mobile robots," *IEEE Trans. Robot. Autom.*, vol. 15, no. 2, p. 219, Oct. 1999. [Online]. Available: https://www.researchgate.net/publication/3298920
[16] J. Steckel and H. Peremans, "Acoustic flow-based control of a mobile platform using a 3D sonar sensor," *IEEE Sensors J.*, vol. 17, no. 10, pp. 3131–3141, May 2017. Online]. Available: http://ieeexplore.ieee.org/document/7888490/
[17] P. C. Hansen, *Discrete Inverse Problems*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, Jan. 2010. [Online]. Available: http://epubs.siam.org/doi/book/10.1137/1.9780898718836
[18] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, "Deep convolutional neural network for inverse problems in imaging," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4509–4522, Sep. 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7949028/
[19] M. Gong, X. Jiang, and H. Li, "Optimization methods for regularization-based ill-posed problems: A survey and a multi-objective framework," *Frontiers Comput. Sci.*, vol. 11, pp. 362–391, Jun. 2017.
[20] A. Kirsch, *An Introduction to the Mathematical Theory of Inverse Problems* (Applied Mathematical Sciences), vol. 120. New York, NY, USA: Springer, 2011. [Online]. Available: http://link.springer.com/10.1007/978-1-4419-8474-6
[21] S. Arridge, P. Maass, O. Öktem, and C.-B. Schönlieb, "Solving inverse problems using data-driven models," *Acta Numerica*, vol. 28, pp. 1–174, May 2019.
[22] M. T. McCann, K. H. Jin, and M. Unser, "Convolutional neural networks for inverse problems in imaging: A review," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 85–95, Nov. 2017.
[23] J. Haahr Christensen, S. Hornauer, and S. Yu, "BatVision: Learning to see 3D spatial layout with two ears," 2019, *arXiv:1912.07011*. [Online]. Available: http://arxiv.org/abs/1912.07011
[24] J. Steckel, A. Boen, and H. Peremans, "Broadband 3-D sonar system using a sparse array for indoor navigation," *IEEE Trans. Robot.*, vol. 29, no. 1, pp. 161–171, Feb. 2013. [Online]. Available: http://ieeexplore.ieee.org/document/6331017/
[25] H. Rebecq, R. Ranftl, V. Koltun, and D. Scaramuzza, "Events-to-video: Bringing modern computer vision to event cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 3857–3866, Apr. 2019. [Online]. Available: http://arxiv.org/abs/1904.08298
[26] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. 18th Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, vol. 9351, 2015, pp. 234–241. [Online]. Available: http://link.springer.com/10.1007/978-3-319-24574-4_28
[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778. Online]. Available: http://image-net.org/challenges/LSVRC/2015/
[28] A. Paszke *et al.*, "PyTorch: An imperative style, high-performance deep learning library," 2019, *arXiv:1912.01703*. [Online]. Available: http://arxiv.org/abs/1912.01703
[29] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proc. 3rd Int. Conf. Learn. Represent.*, San Diego, CA, USA, Dec. 2015, pp. 1–15. [Online]. Available: http://arxiv.org/abs/1412.6980
[30] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004. [Online]. Available: http://www.cns.nyu.edu/~lcv/ssim/

NIELS BALEMANS received the bachelor's and master's degrees in electronics and ICT engineering from the University of Antwerp, in 2018 and 2019, respectively. Since 2019, he has been a Researcher. He is currently a Ph.D. Researcher with the CoSys-Lab, Flanders Make Strategic Research Centre, Lommel, Belgium, and the IDLab, imec, Leuven, Belgium. Both research groups are part of the University of Antwerp, Belgium. His research interests include machine learning, robotics, computer vision, and navigation algorithms.

PETER HELLINCKX received the master's degree in computer science and the Ph.D. degree in science from the University of Antwerp, in 2002 and 2008, respectively. He is currently a Professor with the IDLab, imec Research Team, University of Antwerp. He is also the Head of the Department of Electronics-ICT, and the Head of AI with the IDLab-Antwerp, where he initiated the postgraduate in the Internet of Things. He also supervises 16 Ph.D. students, three post-docs, and a development team in the field of distributed artificial intelligence. He is also teaching third year bachelor courses advanced programming techniques, artificial intelligence, and distributed systems, and the master courses the IoT distributed embedded software and computer graphics. He is also Co-Founder of the spin-offs Hysopt, Hi10, and Digitrans. His research interests include distributed artificial intelligence for the IoT and cyber physical systems with as main application domains: autonomous driving/shipping, logistics, mobility, Industry 4.0, and smart cities. In this field, he is also a reviewer in many scientific project evaluation commissions, both on a national and an international level.

JAN STECKEL received the degree in electronic engineering from Karel de Grote University College, Hoboken, in 2007, and the Ph.D. degree from the Active Perception Laboratory, University of Antwerp, in 2012, with a dissertation titled "Array processing for in-air sonar systems—drawing inspirations from biology." During this period, he developed state-of-the-art sonar sensors, both biomimetic and sensor-array based. He pursued industrial exploitation of the patented 3D array sonar sensor which was developed in collaboration during his Ph.D. In 2015, he became a tenure track Professor with the Constrained Systems Laboratory, University of Antwerp, where he researches sensors, sensor arrays, and signal processing algorithms, using an embedded and constrained systems approach. During his postdoctoral training, he was an Active Member of the Centre for Care Technology, University of Antwerp, where he was in charge of various healthcare-related projects concerning novel sensor technologies.