IEEE Transactions on Robotics (T-RO) paper, presented at ICRA 2026, Vienna, Austria. Cite as T-RO paper.

Privacy–Preserving Robotic Perception for Object Detection in Curious Cloud Robotics

Michele Antonazzi, Matteo Alberti, Alex Bassot, Matteo Luperto, Nicola Basilico

***Abstract*—Cloud robotics allows low–power robots to perform computationally intensive inference tasks by offloading them to the cloud, raising privacy concerns when transmitting sensitive images. Although end–to–end encryption secures data in transit, it does not prevent misuse by inquisitive third–party services since data must be decrypted for processing. This paper tackles these privacy issues in cloud–based object detection tasks for service robots. We propose a co–trained encoder–decoder architecture that retains only task–specific features while obfuscating sensitive information, utilizing a novel weak loss mechanism with proposal selection for privacy preservation. A theoretical analysis of the problem is provided, along with an evaluation of the trade–off between detection accuracy and privacy preservation through extensive experiments on public datasets and a real robot.**

I. Introduction

Robotic systems increasingly require advanced computational capabilities to handle the intensive inference demands of modern deep learning architectures. These architectures allow robots to autonomously perform tasks such as perception, navigation, and planning, but they also pose challenges in terms of resource efficiency and real–time performance. Cloud robotics is an increasingly relevant paradigm since it enables one to engineer architectures in which the robot collects data, while inference is carried out remotely, after transmission [1]–[3]. This paradigm allows dealing with the typical conflict between low–powered robots and high–demand deep neural networks' (DNN) inference [4]. Advanced solutions for robotic software systems, as the recent examples of [5], [6] on orchestration platforms for mixed–critical robotic applications and fog robotics, are setting the stage for low–powered robotic platforms to access the extensive computational resources offered by the cloud. An increasingly popular solution is to rely on third–party cloud–based inference services [7].

Service robots are one class of robotic systems that can clearly benefit from this paradigm. These robots often operate in human–centric environments, performing tasks such as domestic assistance, healthcare, and logistics [8]. They are usually equipped with high–resolution cameras and acquire privacy–sensitive images [9], [10]. When service robots rely upon a *TaskNet*, i.e., a third–party cloud–based inference service that performs a target task, privacy is a major issue.

The work of [11] defines privacy preservation for data–streams acquired by service robots with two key requirements: minimizing the risk of exposing human–interpretable images of the robot's operational environment and limiting the information that enables their reconstruction by a third party. End–to–end encryption, a widely used technique to prevent man–in–the–middle attacks, is ineffective in our scenario because the remote inference engine must access the decrypted plain data (both for training and inference), leaving no technological safeguards to potential leaks or misuses by a *curious* untrustworthy end–point. A renowned example of this critical concern is the case of [12], where privacy–sensitive data, depicting owners of vacuum cleaner robots in their homes, were transmitted to the company's servers using secure channels but later leaked by employees.

Against this threat, we need to fulfill two competing objectives. On the one hand, the robot's data stream transmitted from a trusted environment should be obfuscated to resemble noise to a curious observer and be robust against adversarial post–processing for reconstruction. On the other hand, obfuscated images should preserve enough meaningful features to minimize the task–performance gap between a trusted service (that can access the plain images) and a curious one working with obfuscated data. Furthermore, privacy solutions should operate online on the robot, with computational and latency constraints.

In this work, we focus on a particularly relevant task that the robot has to perform, and that highly benefits the cloud robotics scenario, which is object detection [3], [4], [13]. Object detection is widely used to assist core robotic tasks such as semantic mapping [14]–[17] and localization [18], navigation [19], or human–robot interaction [20]. Nevertheless, limited computational capabilities and energy–preserving requirements make it difficult to run in real–time large object–detection models directly on mobile robots. We consider the distributed architecture of Fig. 1: a robot collects images using its onboard camera. Data are processed by a lightweight encoder–decoder *obfuscator* and then sent to a remotely–deployed TaskNet for object–detection. This network is an off–the–shelf model trained on plain images that performs the task. The obfuscator hence safeguards the data against curious attackers with access to the cloud.

Recent works have pointed out how DNNs can be trained to distill features that contain only the necessary information to effectively solve a given task, without revealing any sensible information once the image is reconstructed [21]. A relevant approach is described in [22], wherein it is shown that an encoder–decoder, when co–trained with a TaskNet, extracts a bottleneck representation of the perceptual data that, although not interpretable by humans, can be effectively used for image classification. In this work, we demonstrate that the same approach cannot be applied to object detection, since this task requires a more feature–rich representation. This is

All authors are with the Department of Computer Science, University of Milan, Milano, Italy.

IEEE Transactions on Robotics (T-RO) paper, presented at ICRA 2026, Vienna, Austria. Cite as T-RO paper.

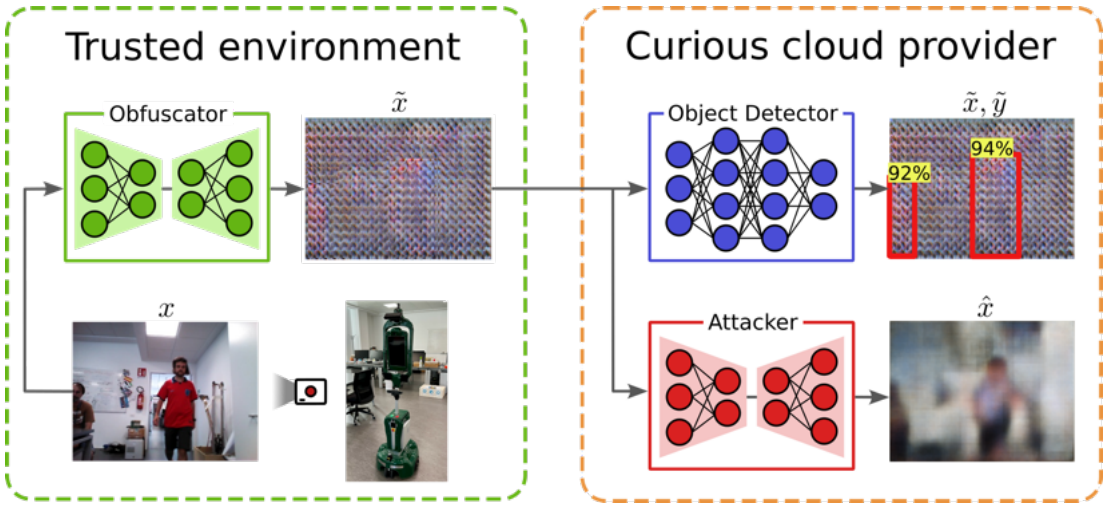


Fig. 1. A general overview of the cloud robotic scenario we consider. A robot collects visual perceptions (x) from its working environment (bottom left) that are obfuscated by a local encoder–decoder, obtaining privatized images $\tilde{x}$ (top left). Its goal is to remove sensitive information from images while maintaining the key features for executing a perception task. Then, the obfuscated robot's perceptions are sent remotely to a curious cloud provider running a TaskNet (pre–trained on plain images) that performs object detection (top right). In our case, the TaskNet extracts a set of bounding boxes $\tilde{Y}$ containing people. In the meanwhile, a malicious actor intercepts the obfuscated images $\tilde{x}$ to reconstruct the original perception x with an adversarial encoder–decoder (bottom right).

due to the task's complexity: multiple targets, of unknown size and location, have to be identified in the same image, forcing the encoder–decoder to preserve redundant details, thus compromising privacy. To overcome this, we propose a novel co–training scheme that systematically selects subsets of proposals over which the loss is computed at each back-propagation pass. We refer to this approach as *weak loss via proposal selection*, and we show that it enables the obfuscator to distill task–relevant features, effectively balancing the competing desiderata of object detection and privacy. Without assuming any specific motion model, our approach obfuscates the data stream, thus providing a general–purpose detection–oriented privatization method that is specifically designed for real–world robotic scenarios, deployable in heterogeneous contexts and with different embodiments.

The obfuscated data stream provided by our method appears indistinguishable to noise from a human observer and, more importantly, is robust to adversarial post–processing carried out by an attacker aiming to reconstruct the original data, matching the definition of privacy outlined in [11]. To assess this last property, we test the robustness of our method against a highly capable malicious actor that, operating within the cloud provider (e.g., an employee), performs a Model Inversion Attack (MIA) against collaborative inference [23] aiming at reconstructing the original perceptions from their obfuscated version (see Fig. 1). Our contributions are summarized as follows:

- we theoretically analyze the properties needed by an obfuscator encoder–decoder to obtain both detection and privacy, overcoming the limitations of [22] (Section III);
- we propose a co–training scheme based on a weak loss with proposals selection to extract only the necessary features for object detection while discarding sensitive information (Section IV);
- we investigate the trade–off between object detection performance and privacy implementing an attack model which aims at reconstructing the original data (see Fig. 1);
- we provide an extensive experimental evaluation of our method relying on publicly available datasets for object detection and performing real–word experiments using a robotic platform (Giraff [24]) deployed on the field (Section V).

Our results prove the validity of our approach in striking a balanced trade–off between task performance and privacy for robotic perception in cloud scenarios.

II. Related Works

Preserving privacy in images and video streams has been a widely studied research topic across different fields. This issue is particularly relevant when the data stream is acquired by low–powered robotics platforms in private environments and sent remotely to untrusted curious cloud providers [25]. The work by [10] examines these concerns in robot teleoperation, focusing on identifying key privacy features to be privatized, like object locations and types.

A computationally inexpensive yet effective way to privatize images is by applying filters, such as blurring, pixelating, or mosaicing. Blurring reduces image detail by averaging pixels, pixelating lowers resolution, and mosaicing breaks the image into smaller tiles and rearranges them. However, recent literature demonstrates that these techniques are not robust against deep–learning attack models [26]: the work of [27] proposes a super–resolution method to remove blur from faces in images while [28] shows how generative models can reconstruct pixellated faces.

The recent advancements in machine learning have also facilitated the development of new privacy–preservation techniques [26]. A promising approach is Homomorphic Encryption [29], which aims to allow DNNs to work with both plain and encrypted data. However, these models struggle to work

in real–time as needed by a mobile robot and still exhibit a large performance gap between the use of plain and encrypted data [30].

Another family of approaches aims at masking sensitive information from images (e.g., faces or people's silhouettes). A GAN–based architecture is exploited in [31] to alter people's appearance by generating a synthetic face, while the work of [32] aims at masking the whole person's figure. The method of [33] preserves the high–resolution background of images perceived by a mobile robot, while a person's identity is obfuscated using a low–resolution mask. In [34] authors propose to use a GAN to modify only the face of the user when seen by a social robot, to prevent its identification. Although these methods are promising, introducing privacy only on sensitive parts of an image is not enough, as sensitive data can be sent out due to faults in the system. Also, those methods have high computational requirements and are vulnerable to missing detections: a single misclassification in a data stream might expose a person's identity. Moreover, sensitive information can appear in the background (e.g., a credit card on a table). In this work, we consider this more challenging scenario, seeking to obfuscate all the acquired data: not only people's identities but also their activities and the background scene.

Differential Privacy (DP) is exploited in [35] to prevent face identification by adding noise to slightly alter the face's appearance without complete obfuscation. While DP is robust against single–frame attacks, it is less effective when applied to data streams, as the attacker can exploit the correlation between frames to filter the added noise.

The works of [36], [37] discuss how activity recognition can be performed on privacy–preserving low–resolution videos, as small as 16×12 pixels. Similar findings are reported in [38] for posture classification and in [39] for low–resolution RGB–D data. Despite being promising for privacy preservation, these approaches cannot be used for object detection as low–resolution data compromises the spatial integrity of features.

Our architecture presents some similarities with that of [40]. Such a work focuses on optimizing a DNN for image classification while minimizing the effectiveness of a DNN–based attacker that exploits adversarial reconstruction and classification techniques. Unlike ours, their framework relies on a *trusted* cloud environment, while we are considering a more challenging *untrusted* one. Another related approach is explored in [41], where a distributed perception pipeline is examined. In this setup, edge devices upload images to a cloud–based TaskNet. The study utilizes an autoencoder to erase sensitive information while retaining essential features for task performance. Unlike the work described in [22] and our setting, the proposed encoder–decoder architecture is embedded within the TaskNet backbone, which is divided into two segments: one deployed on the device and the other on the cloud. The system has been evaluated focusing on face and license plate recognition. While similar to our approach, this method results in obfuscation concentrated around edges and textured regions, thereby maintaining privacy–sensitive details in other parts of the image.

The work of [42] uses a Transformers–based approach to extract tokens used for the task of activity recognition in a video stream. Tokens are anonymized through adversarial learning. As the video stream is processed as a whole, the method cannot be used in an online setting like ours.

NinjaDesc [21] is an adversarial learning framework that extracts visual descriptors from images, for the task of matching similar images (e.g., for localization), while limiting an attacker to reconstruct obfuscated ones. To preserve accuracy in matching similar images, NinjaDesc descriptors retain the keypoint locations. As a consequence, patterns of keypoints can be used to identify persons as they represent the scene structure.

This work aims to address the limitations of the aforementioned approaches when applied in scenarios where mobile robots, that operate in private environments, need to send their perceptions remotely to perform object detection. Instead of masking specific privacy–sensitive features (as performed in [31], [33], [34]), our method provides a novel training approach for global obfuscation of the image, removing as many details as possible while enabling the TaskNet to work even with obfuscated images that do not contain privacy–sensitive details. To achieve this, we take inspiration from the work of [22] that aims to extract from images task–relevant feature representation for image classification. We provide theoretical evidence of its limitations in object detection tasks, and we formally define a novel co–training scheme based on weak loss to enable an encoder–decoder (running on the robot) to achieve privacy while preserving detection–oriented features. We assess the effectiveness of our approach using publicly available datasets as well as experiments with a real robot. In addition, we corroborate the robustness of our privatization technique exploiting the well–established Model Inversion Attack (MIA) [23] in its stronger version, where the malicious actor has full access to the encoder–decoder and the TaskNet, as well as their training datasets. The attacker is implemented following the same approach of [41], which we extend by incorporating a more powerful edge–centric loss function.

III. Problem Modeling and Analysis

A. Scenario and Problem Formalization

In the scenario we consider (Fig. 1), the robot collects images $x \in \mathbb{R}^D$, with $D = W \times H \times C$ and TaskNet is assumed to be a differentiable object detector following an architecture that generates a dense set of object proposals (labeled bounding boxes), which are then filtered to produce a small set of final predictions [43]. This type of architecture is widely adopted by many prominent and popular detectors [44], [45]. We assume that the TaskNet's parameters θ_t are given and fixed (for example, because the service provider has trained the DNN from any of the publicly available pre–trained object detectors [43]). We denote by $Y = f(x; \theta_t)$ its dense set of unfiltered proposals for image x, where each proposal is a labeled bounding box with some level of confidence.

Before being transmitted to the cloud, x is processed by a differentiable encoder–decoder DNN which produces an obfuscated version $\tilde{x} \in \mathbb{R}^D$, formally defined as $\tilde{x} = q(x; \theta_e, \theta_d)$, where θ_e and θ_d are the DNN's learnable parameters. We indicate with $z = e(x; \theta_e)$, the low–dimensional

latent representation in $\mathbb{R}^Z$, with $Z \leq D$ being the *bottleneck* dimension, computed by the encoder. The final obfuscated representation is denoted as $\tilde{x} = d(z; \theta_d)$. TaskNet computes objects' locations and classes for the image x by inference on its obfuscated representation $\tilde{x}$. Against this background, we introduce the problem of training our encoder–decoder module so that TaskNet achieves comparable performance when working on $\tilde{x}$ as it does when receiving as input the original image x. At the same time, we seek representations $\tilde{x}$ from which sensible information cannot be extracted. Note that θ_t has been obtained by training TaskNet on plain images x.

The first objective can be achieved by minimizing a *task loss* $\mathcal{L}_{task}(Y, \tilde{Y})$. This loss is proportional to the difference between the dense sets of proposals returned by TaskNet with plain and obfuscated data, respectively. It rewards obfuscated images that preserve task–related features. For the second objective, we introduce a *forward privacy loss*, $\mathcal{L}^{\gg}_{priv}$. Specifically, $\mathcal{L}^{\gg}_{priv}(x, \tilde{x})$ measures the weakness of the representation $\tilde{x}$ against the threat posed by our scenario. This loss is meant to promote representations that suppress features related to sensible information. We use the term "forward" to remark its relationship with the process of privatizing an image x into its obfuscation $\tilde{x}$ that is forwarded to an untrusted environment. Expressing a trade–off between the two objectives with a parameter λ, the learning problem can be formalized as follows.

Problem 1 (Co–training for Perception and Privacy)**.** *Given $\lambda \in [0, 1]$ and a differentiable pre–trained TaskNet $f(\cdot; \theta_t)$, find encoder–decoder bottleneck dimension Z and parameters (θ_e, θ_d) such that, for data distribution $\mathcal{D}$, the following is minimized:*

$$\mathop{\mathbb{E}}_{x \sim \mathcal{D}} \lambda \mathcal{L}_{task}(Y, \tilde{Y}) + (1 - \lambda)\mathcal{L}^{\gg}_{priv}(x, \tilde{x}).$$

Then, we introduce an *attacker*, explicitly formalizing the process of executing our scenario's threat, which is inspired by MIA [23]. The attacker takes as input the image representation $\tilde{x}$ and aims to compute a new representation $\hat{x}$ where sensible information is restored to some degree. In line with the MIA paradigm, we adopt an encoder–decoder architecture for the attacker too, denoted by $(\hat{\theta}_e, \hat{\theta}_d)$. To train such a model, we introduce a *backward privacy loss* $\mathcal{L}^{\ll}_{priv}$, designed to promote the attacker's ability to restore sensible information of x, working backward from its reconstruction $\tilde{x}$, as follows[1].

Problem 2 (Privacy Violation)**.** *Given a DNN $q(\cdot; \theta_e, \theta_d)$, find the encoder–decoder bottleneck dimension $\hat{Z}$ and parameters $(\hat{\theta}_e, \hat{\theta}_d)$ such that, for the data distribution $\mathcal{D}$, the following is minimized: $\mathop{\mathbb{E}}_{x \sim \mathcal{D}} \mathcal{L}^{\ll}_{priv}(x, \hat{x})$.*

Formally defining $\mathcal{L}^{\gg}_{priv}$ and $\mathcal{L}^{\ll}_{priv}$ is no easy task [26]. Given its high variability, subjectivity, and domain dependence, obfuscating or restoring sensible information in an image is difficult to model and encode in a loss function. It could be argued that $\mathcal{L}^{\gg}_{priv}$ and $\mathcal{L}^{\ll}_{priv}$ are related to reconstruction. Thus, $\mathcal{L}^{\gg}_{priv}$ can be defined as inversely proportional to a reconstruction loss, while $\mathcal{L}^{\ll}_{priv}$ can be defined as directly proportional. Following the last implication, we set $\mathcal{L}^{\ll}_{priv} = \mathcal{L}_{rec}$ (where $\mathcal{L}_{rec}$ is a reconstruction loss) as we assume that the attacker of Problem 2 seeks reconstruction because it implies the restoration of any sensible information (the same rationale behind MIA). However, implementing $\mathcal{L}^{\gg}_{priv}$ of Problem 1 as an inverse reconstruction loss is likely to be ineffective: not reconstructing clearly does not always imply adding privacy.

An alternative approach is to set $\lambda = 1$, thereby neglecting the privacy objective represented by $\mathcal{L}^{\gg}_{priv}$, while decreasing the bottleneck dimension Z of the encoder–decoder. The rationale is that it is possible to learn representations which, due to high inner compression, are not human–interpretable once reconstructed, yet can still recall a similar task performance. This approach has been shown to be effective in [22] when TaskNet is a classifier. In the following, we prove that this method cannot be used for object detection tasks providing a theoretical analysis (Section III-B) and confirming it with empirical evidence (Section III-C).

***Our solution**:* We introduce a novel approach to address Problem 1 without the need to explicitly define $\mathcal{L}^{\gg}_{priv}$. Our method is based on the concepts of *weak task loss* and *proposal selection* (Section IV). Specifically, we train the encoder–decoder using a weakened version of the TaskNet's loss, computed over a limited subset of proposals at each iteration. This encourages the encoder–decoder to retain only essential task–relevant features, while discarding irrelevant details (thereby enhancing privacy) without requiring a reduction in the bottleneck size. The proposal selection strategy (detailed in Section IV-A) selects a small number of proposals for each target, with the goal of promoting correct predictions while reducing errors.

B. Detection vs. Privacy: Theoretical Analysis

In [22], the authors propose a task–oriented framework for data compression in robotic applications. The method co–trains a variational autoencoder (VAE) [46] with a TaskNet for classification. Then, the encoder of the VAE is deployed on the robot to produce compact and task–specific representations of visual perceptions that are transmitted to a remote server over a low–bandwidth connection. On the server side, the VAE's decoder recovers the original dimensionality of the compressed perceptions (while maintaining task–related features) that are finally fed into TaskNet for inference. Such a work demonstrates that decreasing the encoder–decoder bottleneck to a dimension Z that is substantially lower than the input's dimensionality yields representations that are not human–interpretable (and so private) since reconstruction features are not needed (so not learned). The authors provide theoretical support for their findings by studying a linearized model of their framework. This kind of analysis is common in the literature (see [47] for a recent example) as studying such simplified models can offer formal (hence explainable) insights into the behavior of more complex ones. In this section, we adopt this approach to examine the trade–offs that characterize

[1]We use the symbol $\gg$ to indicate the action of adding privacy (forward, i.e., from the data source), while $\ll$ to compromise privacy (backward, i.e., from the attacker).

our scenario and justify our proposed method, which we devise and evaluate in the subsequent sections.

Following the same methodology of [22], we model TaskNet as a proposal–based detector, aligning with one of the mainstream approaches in deep neural network architectures for object detection [43]. Proposal–based architectures extract a meaningful representation of the input using a convolutional backbone (such as a Residual Network, ResNet [48]) which is then processed by a dense set of overlapping object–proposal heads with different dimensions and scales that, being uniformly distributed across the image, are responsible of detecting objects. This approach is implemented by the off–the–shelf one–stage and two–stage detectors with a key difference in how proposals are handled. In one–stage detectors like YOLO [45], the object proposals are specified by means of hyperparameters, while, in two–stage detectors such as Faster R–CNN [44], the location and size of object proposals are dynamically computed during inference. In defining a generic proposal–based TaskNet for the linear setting, we explicitly model this structure encompassing a backbone and a set of heads.

Definition 1 (Linear TaskNet for Object Detection)**.** *In the linear setting, we consider a TaskNet for Object Detection composed of (i) a backbone defined as a full–rank matrix $K \in \mathbb{R}^{S\times D}$ with $S \leq D$ and (ii) a set of object proposals $\mathcal{H} = \{H_1, \ldots, H_n\}$, in which each $H_i \in \mathbb{R}^{|\mathcal{C}|\times S}$ is a classification head that maps elements of Kx to a tuple of scores for object categories in $\mathcal{C} = \{c_1, c_2, \ldots, \texttt{background}\}$, with $|\mathcal{C}| \leq S$.*

In the above definition, element (i) models the fact that the TaskNet compresses inputs $x \in \mathbb{R}^D$ using the backbone K to produce a meaningful representation of the image, where the features are informative and not redundant. Element (ii) describes that the compressed input $Kx \in \mathbb{R}^S$ is processed by multiple classification heads to detect objects. The rationale behind these two properties is that to perform well, object detectors must retain relevant image features and have a large and dense set of classification heads covering multiple regions of the image. In real object detectors, $|\mathcal{H}|$ is typically in the order of thousands to ensure an accurate and comprehensive object localization [43]. In our derivations, similar to [22], we represent the image x as a one–dimensional vector. We assume each head H_i focuses on a consecutive segment of Kx, ranging from the u-th to the v-th component, where $u, v \in [1, S]$ and $u \leq v$. These assumptions are without loss of generality with respect to the 3D case and facilitate the formal derivations we provide in the following. Since each H_i is a generic matrix, this can be represented by assuming all columns of H_i are zero except for those ranging from column u to column v. In this setup, the task loss for co–training can be defined as:

$$\mathcal{L}_{task}(x) = \sum_{i=1}^{n} \left\| H_i K x - H_i K B A x \right\|_2^2 \tag{1}$$

where $A \in \mathbb{R}^{Z\times D}$ and $B \in \mathbb{R}^{D\times Z}$ are the encoder–decoder matrices with bottleneck dimension Z. Intuitively, this loss function quantifies the difference between TaskNet's performance on plain inputs x and their corresponding obfuscated versions $\tilde{x}$. Within this framework, we can derive the following result.

Theorem 1 (Linear Detection–Aware Compression)**.** *Consider a TaskNet as per Definition 1, with a bottleneck $K \in \mathbb{R}^{S\times D}$, and encoder–decoder matrices $A \in \mathbb{R}^{Z\times D}, B \in \mathbb{R}^{D\times Z}$. Then, when setting the bottleneck dimension $Z = rank(K) = S$ it is possible to achieve $\mathcal{L}_{task}(x) = 0$ for any x and for any $\mathcal{H}$. Moreover, when setting $Z < rank(K)$ there always exist $\mathcal{H}$ and x such that $\mathcal{L}_{task}(x) > 0$.*

Proof. With a bottleneck dimension $Z = rank(K)$ we can apply the same technique of [22]. Consider the compact singular value decomposition (SVD) of $K = U\Sigma V^\top$, where $U \in \mathbb{R}^{S\times S}$ unitary, $\Sigma \in \mathbb{R}^{S\times S}$, and $V^\top \in \mathbb{R}^{S\times D}$ semi–unitary. Solving Eq. 1 for zero is feasible for any input x and any set $\mathcal{H}$ by assigning $A^\top = B = V$. Given that $V^\top V = I_S$, it follows that

$$\begin{aligned}\mathcal{L}_{task}(x) &= \sum_{i=1}^{n} \Big\| H_i \Big(\underbrace{U\Sigma V^\top}_{K}(x - \underbrace{VV^\top}_{BA} x) \Big) \Big\|_2^2 \\ &= \sum_{i=1}^{n} \Big\| H_i \Big(U\Sigma (V^\top x - \underbrace{V^\top V}_{I_S} V^\top x) \Big) \Big\|_2^2 = 0.\end{aligned}$$

Now, we show that for any $Z < rank(K)$ there exists $\mathcal{H}, x$ such that $\mathcal{L}_{task}(x) > 0$. Suppose to use an encoder–decoder given by full–rank matrices $\overline{A} \in \mathbb{R}^{Z\times D}$ and $\overline{B} \in \mathbb{R}^{D\times Z}$ with $Z < rank(K)$. Now consider an $\mathcal{H}$ where each head has the form of $H_i = \begin{bmatrix} 0 & I_{|\mathcal{C}|} & 0 \end{bmatrix} \in \mathbb{R}^{|\mathcal{C}|\times S}$. Assume that the identity matrix $I_{|\mathcal{C}|}$ is shifted differently in different heads, such that $\begin{bmatrix} H_1^\top, \ldots, H_n^\top \end{bmatrix}^\top$ has no all–zeros columns. At an intuitive level, such an $\mathcal{H}$ represents a straightforward set of heads that ensures no feature computed by the backbone is entirely disregarded. If the set of heads has this property, then the only way to achieve zero task loss is to have $Kx = K\overline{B}\overline{A}x$ for any input $x \in \mathbb{R}^D$. This means we must have $K = K\overline{BA}$, but this is impossible since $rank(K\overline{BA}) \leq Z < rank(K)$. □

The above result provides an optimal solution to Problem 1 when $\lambda = 1$ through a factorization of the backbone K. In this solution, regardless of the classification heads and the input x, the dimensionality of the encoder–decoder bottleneck must be at least as large as the rank of K.

In the following, we highlight an important relation between the rank of the encoder–decoder matrices and the reconstruction quality that, as discussed in Section III-A, can be mapped to the objective of the attacker defined in Problem 2. To do this, we define a dataset as a full–rank matrix $X \in \mathbb{R}^{D\times N}$ containing $N \geq D$ examples. With a slight notation overload, we reformulate the task loss from Eq. 1 so that it applies to a dataset X rather than a single instance x and exploit the Cauchy–Schwarz inequality to derive an upper bound on it. Let X_c denote the c–th column of X, representing the c–th example in the dataset, and use $\|\cdot\|_F$ to indicate the Frobenius norm, then

$$\begin{aligned}\mathcal{L}_{task}(X) &= \sum_{i=1}^{n}\sum_{c=1}^{N}\left\|H_i(KX_c - KBAX_c)\right\|_2^2 \\ &= \sum_{i=1}^{n}\left\|H_i(KX - KBAX)\right\|_F^2 \\ &= \sum_{i=1}^{n}\left\|H_i(K - KBA)X\right\|_F^2 \\ &\leq \sum_{i=1}^{n}\left\|H_i\right\|_F^2\left\|K - KBA\right\|_F^2\left\|X\right\|_F^2.\end{aligned}$$

This formulation provides a way to understand changes in task loss through the use of encoder–decoder matrices with ranks $Z < S$. The upper bound indicates that a method to limit $\mathcal{L}_{task}$, without taking into account both the classification heads $\mathcal{H}$ and the dataset X (which, in our setting, we don't control), involves approximating the backbone K with low–rank encoder–decoder matrices. This can be accomplished by solving the following optimization problem:

$$\begin{aligned}&\arg\min_{A,B}\left\|K - KBA\right\|_F^2 \qquad \text{subject to} \\ &rank(A) = rank(B) = Z < S.\end{aligned} \tag{2}$$

For the Eckart–Young theorem [49], the solution of (2) is to have $KBA = U_Z\Sigma_Z V_Z^\top$, which is the Z–truncated SVD of K. In this context, setting $B = V_Z$ and $A = V_Z^\top$ is an optimal solution since

$$\begin{aligned}KBA &= U\Sigma V_S^\top V_Z V_Z^\top \\ &= \begin{bmatrix}U_Z & U_{S-Z}\end{bmatrix}\begin{bmatrix}\Sigma_Z & 0 \\ 0 & \Sigma_{S-Z}\end{bmatrix}\begin{bmatrix}V_Z^\top \\ V_{S-Z}^\top\end{bmatrix}V_Z V_Z^\top \\ &= \begin{bmatrix}U_Z\Sigma_Z & U_{S-Z}\Sigma_{S-Z}\end{bmatrix}\begin{bmatrix}I_Z \\ 0\end{bmatrix}V_Z^\top \\ &= U_Z\Sigma_Z V_Z^\top.\end{aligned}$$

In other words, for a TaskNet's backbone K with rank S, optimizing the task loss in a heads– and dataset–independent manner using encoder–decoder matrices with rank $Z < S$ means setting $A = V_Z^\top$ and $B = V_Z$, where V_Z is obtained by removing the last $S-Z$ columns of V_S extracted according to Theorem 1. We exploit these derivations to provide a result that shows how, in our setting, diminishing the encoder–decoder's rank through bottleneck compression offers privacy to the robot's perceptions by hindering the attacker's reconstruction capability.

Theorem 2 (Compression vs. Reconstruction)**.** *Let $X \in \mathbb{R}^{D\times N}$ be a full–rank dataset matrix with $N \geq D$ examples and consider a TaskNet with backbone K according to Definition 1. Consider a reconstruction loss $\mathcal{L}_{rec}^{Z} = \|X - V_Z V_Z^\top X\|_F^2$ where encoder–decoder matrices V_Z and $V_Z^\top$ are obtained by removing the last $S-Z$ columns from V_S obtained from K according to Theorem 1. Then, for any $Z_1 < Z_2 \leq S$, we have $\mathcal{L}_{rec}^{S}(X) \leq \mathcal{L}_{rec}^{Z_2}(X) < \mathcal{L}_{rec}^{Z_1}(X)$ for all X.*

Proof. The matrix $V_S \in \mathbb{R}^{D\times S}$ derived from Theorem 1 consists of Z orthogonal columns. For $Z < S$, the matrix V_Z is constructed by removing columns from V_S according to the resolution of (2). Define $V_D = \begin{bmatrix}V_S & V_{D-S}\end{bmatrix}$, where V_{D-S} provides $D - S$ additional columns to ensure V_D is unitary. This is achieved by composing V_{D-S} with columns that, together with those of V_S, create an orthogonal basis for the vector space $\mathbb{R}^D$. For any $Z < D$, we can write

$$V_Z V_Z^\top = V_D\begin{bmatrix}I_Z & 0 \\ 0 & 0\end{bmatrix}V_D^\top.$$

Thus, we can expand the reconstruction loss as

$$\begin{aligned}\mathcal{L}_{rec}^{Z}(X) &= \left\|X - V_Z V_Z^\top X\right\|_F^2 \\ &= \left\|I_D X - V_D\begin{bmatrix}I_Z & 0 \\ 0 & 0\end{bmatrix}V_D^\top X\right\|_F^2 \\ &= \left\|V_D V_D^\top X - V_D\begin{bmatrix}I_Z & 0 \\ 0 & 0\end{bmatrix}V_D^\top X\right\|_F^2 \\ &= \left\|V_D(V_D^\top X - \begin{bmatrix}I_Z & 0 \\ 0 & 0\end{bmatrix}V_D^\top X)\right\|_F^2 \\ &= \left\|V_D^\top X - \begin{bmatrix}I_Z & 0 \\ 0 & 0\end{bmatrix}V_D^\top X\right\|_F^2 \\ &= \left\|[V_D^\top X]_{Z:D}\right\|_F^2,\end{aligned}$$

where $[V_D^\top X]_{Z:D} \in \mathbb{R}^{(D-Z)\times N}$ is the sub–matrix of $V_D^\top X$ containing the last $D - Z$ rows. With the reconstruction loss expressed in this form, it is easy to see that

$$Z_1 < Z_2 \implies \underbrace{\left\|[V_D^\top X]_{Z_2:D}\right\|_F^2}_{\mathcal{L}_{rec}^{Z_2}(X)} < \underbrace{\left\|[V^\top X]_{Z_1:D}\right\|_F^2}_{\mathcal{L}_{rec}^{Z_1}(X)},$$

which can be generalized as:

$$\mathcal{L}_{rec}^{S}(X) < \mathcal{L}_{rec}^{S-1}(X) < \ldots < \mathcal{L}_{rec}^{1}(X) < \left\|X\right\|_F^2. \qquad \square$$

The above results shed some light on a significant trade–off of our robotic scenario in Fig. 1. In this context, we have the ability to design the robot's perception module, yet we do not have control over the TaskNet responsible for object detection, as we depend on an external provider. TaskNet runs in the cloud, and the robot can interface with it by uploading sensory input. Our aim is to diminish an attacker's ability to reconstruct the robot's sensory data while maintaining the performance level TaskNet achieves with direct images. We introduce an obfuscator with an encoder–decoder to process the sensory inputs, and we co–train it using the TaskNet supervision to obtain the same detection performance as using plain images, i.e., with the objective of minimizing Eq. 1. However, preserving TaskNet's performance requires the obfuscator to avoid excessive compression of the robot's perceptions. If its bottleneck matches the TaskNet's backbone, full performance recovery is achievable (Theorem 1, first part). However, choosing a smaller bottleneck might lead to performance degradation. This degradation may occur if the TaskNet's components or the inputs themselves (which in this scenario are beyond our control) are such that reducing the bottleneck increases the task loss (Theorem 1, second part). In contrast, obstructing an attacker's ability to reconstruct the robot's sensory data from the obfuscated images could be facilitated by opting for a smaller bottleneck in the obfuscator (Theorem 2). Thus, safeguarding the robot's perceptual

IEEE Transactions on Robotics (T-RO) paper, presented at ICRA 2026, Vienna, Austria. Cite as T-RO paper.

TABLE I
COMPARISON OF PEOPLE DETECTION PERFORMANCE AS A FUNCTION OF THE ENCODER–DECODER BOTTLENECK DIMENSION. VAE_Z REFERS TO THE SETUP FROM [22], WHICH USES A VARIATIONAL AUTOENCODER. EDO REPRESENTS OUR APPROACH, WHERE THE VAE IS REPLACED BY AN ENCODER–DECODER OBFUSCATOR. TN (TASKNET ON PLAIN IMAGES) DENOTES THE UPPER–BOUND REFERENCE PERFORMANCE. FOR BOTH EDO AND VAE, RESULTS ARE SHOWN IN DECREASING ORDER OF BOTTLENECK SIZE. PERFORMANCE ON IN–DISTRIBUTION (COCO) AND OUT–OF–DISTRIBUTION (PASCAL VOC) DATA HIGHLIGHTS HOW REDUCING THE BOTTLENECK DIMENSION PROGRESSIVELY DEGRADES TASKNET'S REFERENCE PERFORMANCE.

Setting	COCO		Pascal VOC	
	AP ↑	AP_{50} ↑	AP ↑	AP_{50} ↑
TaskNet (TN)	59	87	55	86
EDO	47	76	39	68
VAE_{256}	9	24	5	16
VAE_{128}	8	22	5	15
VAE_{64}	7	21	4	14

data using obfuscation while simultaneously achieving task performance with obfuscated images drives the design of the robot's perception processing in conflicting directions. In the following section, we empirically validate this intuition within a realistic setting.

C. Detection vs. Privacy: Empirical Evaluation

We compare the setting of [22], which uses a variational autoencoder (VAE) [46], against a setup in which we replace the VAE with a convolutional encoder–decoder. The key difference between the two is the bottleneck compression: while the VAE encodes input data in a flattened inner representation with dimension Z, the encoder–decoder maps images with dimension $D = (W \times H \times 3)$ to a 2D inner space with $Z = \left(\frac{W}{16} \times \frac{H}{16} \times 1024\right)$, resulting in an adaptive and substantially larger bottleneck dimension. We compare the encoder–decoder with multiple variational autoencoder configurations (named VAE_Z) changing the bottleneck dimension Z on the task of people detection. For the remainder of the paper, we use the encoder–decoder architecture as the main setting, named Encoder–Decoder Obfuscator (EDO). For this evaluation, we consider the task of people detection and implement the TaskNet with a Faster R–CNN [44] re–trained on the `person` object category of COCO dataset [50]. We perform testing using COCO and Pascal VOC 2012 [51] validation sets. Object detection performances are quantified using the AP and AP_{50} of COCO [50]. AP is the area under the precision/recall curve averaged over multiple IoU thresholds for the correct detections (0.5 to 0.95 in steps of 0.05), while AP_{50} considers only the case where the IoU threshold is 0.5. (The specifics of this preliminary evaluation's experimental setup mirror those of our primary campaign and are thoroughly outlined in Section V-A.)

The results reported in Table I show that the VAE substantially degrades the TaskNet's detection performance. TaskNet's performance on plain images, labeled TN, serves as the upper bound reference value. When the images from the COCO benchmark are used, VAE–based architectures have significantly lower performance with respect to TN. More precisely, AP and AP_{50} are only $\approx 15\%$ and 27% of the values reached by TaskNet with plain images (TN). This is caused by the limited bottleneck dimension Z of the VAE, which is unable to encode a dense and uniformly distributed feature representation for precise and dense object detection. Conversely, as suggested by Theorem 1, a larger bottleneck allows us to fill the gap between the performance of the TaskNet with plain (TN) and obfuscated images (EDO). When the encoder–decoder is used in place of the VAE, higher performances are retained. TaskNet, when using images $\tilde{x}$ obfuscated by the EDO, achieves approximately 80% of the AP and 87% of the AP_{50} obtained by TN on the COCO dataset. These results hold true even in out–of–distribution settings. Specifically, on the Pascal VOC dataset, the EDO reaches about 70% of TN's reference value of AP and 79% of its AP_{50}, while VAE has a significant drop in performance, retaining only about 9% and 18% of AP and AP_{50}, respectively.

These findings demonstrate that, in accordance with Theorem 1, achieving optimal task performance requires a bottleneck size that is (large and) nearly equal to the size of the TaskNet's backbone. Conversely, to obfuscate sensitive information, in line with Theorem 2, additional compression of the encoder–decoder bottleneck is required. This is also confirmed by the examples in Fig. 2: the size of the EDO's bottleneck, being near the input dimensionality, preserves privacy–related features compromising identity preservation in both $\tilde{x}$ and $\hat{x}$.

IV. WEAK LOSS VIA PROPOSAL SELECTION

The above analysis indicates that within the conventional learning framework for object detectors utilizing dense proposal sets, modifying the size of the encoder–decoder bottleneck alone does not suffice to jointly achieve effective perception and privacy. Theorem 1 demonstrates that setting the encoder–decoder considering only the backbone K is a convenient solution when the bottleneck dimension $Z = rank(K)$. However, setting $Z < rank(K)$ may lead to degradation of task performance. At the same time, Theorem 2 suggests that lowering Z results in a bad reconstruction by the attacker (that is, good privacy). Considering Theorem 1, we propose a solution designed to jointly minimize task loss and achieve privacy. Our method extends the learning framework considered in our analysis by adding a proposal selection mechanism, limiting the proposals employed by the learning algorithm during the computation of the loss function. The concept of proposal selection is well known, but with popular off–the–shelf object detectors it generally functions as a downstream filtering process applied during inference time to improve bounding box accuracy. Common techniques include confidence thresholding and non–maximum suppression [43].

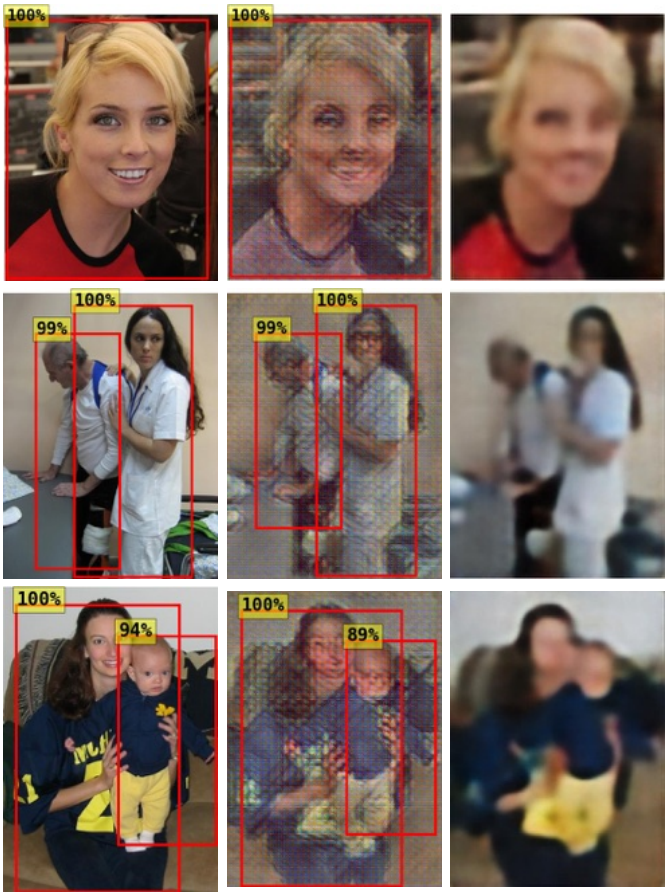
We formulate our approach in linear settings considering the framework introduced in Section III-A. Our goal is to find sets of classification heads $\mathcal{H}$ that satisfy the following property.

Property 1. *Given a TaskNet according to Definition 1,*

$$\exists A \in \mathbb{R}^{Z \times D}, B \in \mathbb{R}^{D \times Z} \text{ with } Z < S \mid \forall x \, \mathcal{L}_{task}(x) = 0.$$

The following result characterizes any TaskNet with Property 1.

IEEE Transactions on Robotics (T-RO) paper, presented at ICRA 2026, Vienna, Austria. Cite as T-RO paper.



Raw image x, y | Obfuscation $\tilde{x}, \tilde{y}$ | Reconstruction $\hat{x}$

Fig. 2. (Left) Original images from the Pascal VOC dataset, (middle) obfuscated outputs from our EDO, and (right) reconstructions generated by the attacker. Red bounding boxes indicate TaskNet detections y and $\tilde{y}$ with confidence scores $\sigma(y), \sigma(\tilde{y}) \geq 0.75$, filtered using Non–Maximum Suppression (NMS) with an IoU threshold of 0.5. The examples show that when the bottleneck size is close to the input dimensionality, the obfuscation retains substantial visual information.

Theorem 3 (Compression for Detection and Privacy)**.** *Given a TaskNet with backbone $K \in \mathbb{R}^{S\times D}$ and $\mathcal{H} = \{H_1, \dots, H_n\}$ as per Definition 1, and considering the matrix $M \in \mathbb{R}^{n|\mathcal{C}|\times D}$ defined as*

$$M \coloneqq \begin{bmatrix} H_1 \\ \vdots \\ H_n \end{bmatrix} \cdot K = \begin{bmatrix} H_1 K \\ \vdots \\ H_n K \end{bmatrix},$$

we have that the TaskNet has Property 1 $\iff$ $rank(M) < S$.

Proof. We separately prove the two directions of ($\iff$).

($\impliedby$) : By construction, we have that

$$H_i K = I_i M \in \mathbb{R}^{|\mathcal{C}|\times D},$$

where $I_i = \begin{bmatrix} 0 & I_{|\mathcal{C}|} & 0 \end{bmatrix} \in \mathbb{R}^{|\mathcal{C}|\times n|\mathcal{C}|}$ in which the identity matrix $I_{|\mathcal{C}|}$ is positioned from column $(i-1)|\mathcal{C}|+1$ to column $i|\mathcal{C}|$ to extract the i^{th} head from M. Now, considering the compact SVD of the matrix $M = U\Sigma V^T$ where $U \in \mathbb{R}^{n|\mathcal{C}|\times Z}$, $\Sigma \in \mathbb{R}^{Z\times Z}$ and $V^T \in \mathbb{R}^{Z\times D}$, with $rank(M) = Z < S$ and setting $\overline{B} = V$ and $\overline{A} = V^T$ produces zero task loss for any input x:

$$\begin{aligned}
\mathcal{L}_{task}(x) &= \sum_{i=1}^{n} \left\| H_i K x - H_i K \overline{BA} x \right\|_2^2 \\
&= \sum_{i=1}^{n} \left\| H_i K x - I_i M \overline{BA} x \right\|_2^2 \\
&= \sum_{i=1}^{n} \left\| H_i K x - I_i M V V^\top x \right\|_2^2 \\
&= \sum_{i=1}^{n} \left\| H_i K x - I_i U \Sigma \underbrace{V^\top V}_{I_Z} V^\top x \right\|_2^2 \\
&= \sum_{i=1}^{n} \left\| H_i K x - I_i U \Sigma V^\top x \right\|_2^2 \\
&= \sum_{i=1}^{n} \left\| H_i K x - I_i M x \right\|_2^2 \\
&= \sum_{i=1}^{n} \left\| H_i K x - H_i K x \right\|_2^2 = 0.
\end{aligned}$$

($\implies$) : Notice that, in general, $rank(M) \leq S$. Thus, suppose by contradiction that $rank(M) = S$ and there exists a couple of matrices $\overline{A} \in \mathbb{R}^{Z\times D}, \overline{B} \in \mathbb{R}^{D\times Z}$ with rank $Z < S$ such that $\mathcal{L}_{task}(x) = 0, \forall x \in \mathbb{R}^D$. This implies that $H_i K = H_i K \overline{BA}$ for all i, that is true iff $M = M\overline{BA}$. This is in contradiction since $rank(M\overline{BA}) \leq Z < S = rank(M)$. □

According to Theorem 3, we can guarantee Property 1 by reducing the rank of the matrix M obtained by stacking the product between each head and the backbone K. With this framework, the following corollaries report two easy methods to select sets of classification heads to ensure $rank(M) < S = rank(K)$, thus allowing the reduction of the bottleneck dimension to enhance privacy maintaining $\mathcal{L}_{task} = 0$.

Corollary 1 (Weak Set)**.** *$\forall \mathcal{H}$ such that $|\mathcal{H}| < \frac{S}{|\mathcal{C}|}$ the TaskNet satisfies Property 1.*

Proof. Given that $M \in \mathbb{R}^{n|\mathcal{C}|\times S}$ and $|\mathcal{H}| = n < \frac{S}{|\mathcal{C}|}$, then $rank(M) \leq n|\mathcal{C}| < S$. For Theorem 3, the TaskNet satisfies Property 1. □

Corollary 2 (Partial Set)**.** *$\forall \mathcal{H} = \{H_1, \dots, H_n\}$ for which $\exists\, j \in \{1, \dots, S\} \mid \forall H_i \in \mathcal{H}$ the j^{th} column is 0 the TaskNet satisfies Property 1.*

Proof. If the j^{th} column of all heads is set to 0 the rank of the matrix obtained by stacking all the $H \in \mathcal{H}$ cannot exceed $S-1$. Given this, we can argue that

$$rank(M) = rank\left(\begin{bmatrix} H_1 \\ \vdots \\ H_n \end{bmatrix} \cdot K \right) \leq S - 1,$$

which implies that the TaskNet satisfies Property 1 for Theorem 3. □

Corollaries 1 and 2 devise two classes of sets of classification heads to ensure Property 1: *Weak* and *Partial*. The former suggests to lower the cardinality of $\mathcal{H}$ at a value less than the

IEEE Transactions on Robotics (T-RO) paper, presented at ICRA 2026, Vienna, Austria. Cite as T-RO paper.

Algorithm 1 Co–Training with Proposal Selection

Input:
- $f(\,\cdot\,;\theta_t)$: a fixed and pre–trained object detector
- $q(\,\cdot\,;\theta_e^0,\theta_d^0)$: an encoder–decoder with random weights
- $\mathcal{D}=\{(x_i,Y_i)\}_{i=1}^{|D|}$: training dataset
- N_{iter}: the number of training rounds
- p, n: the number of positive and negative proposals
- $\bar{\rho}$: the IoU threshold

Output: the trained parameters θ_e^N, θ_d^N

1: $\theta_e^0, \theta_d^0 \leftarrow RandInit()$
2: **for** $\tau \leftarrow 0$ **to** N_{iter} **do**
3: $\quad(x, Y) \sim \mathcal{D}$
4: $\quad\tilde{Y} = f(q(x;\theta_e^\tau,\theta_d^\tau);\theta_t)$
5: $\quad\tilde{Y}^{\text{SEL}} = \textsc{SelectProposals}(Y,\tilde{Y},p,n)$
6: $\quad\theta_e^{\tau+1},\theta_d^{\tau+1} \leftarrow \textsc{BackProp}\big(\mathcal{L}_{\textit{TaskNet}}(Y,\tilde{Y}^{\text{SEL}}),\theta_e^\tau,\theta_d^\tau,\theta_t\big)$
7: **end for**
8: **procedure** $\textsc{SelectProposals}(Y_{GT},Y_t,p,n)$
9: $\quad S \leftarrow \emptyset$
10: $\quad$**for all** $g \in Y_{GT}$ **do**
11: $\quad\quad P \leftarrow \{t \in Y_t \mid \arg\max_{y\in Y_{GT}} \rho(t,y) = g\}$
12: $\quad\quad P^{\text{SEL}} \leftarrow \{t_1 \dots t_p \mid t_i \in P, \rho(t_i,g) \geq \rho(t_j,g), \forall j \geq i\}$
13: $\quad\quad N \leftarrow \{t \in Y_t \setminus P \mid \arg\max_{y\in Y_{GT}} \rho(t,y) = g, \rho(t,g) < \bar{\rho}\}$
14: $\quad\quad N^{\text{SEL}} \leftarrow \{t_1,\dots,t_n \mid t_i \in N, \sigma(t_i) \geq \sigma(t_j), \forall j \geq i\}$
15: $\quad\quad S \leftarrow S \cup P^{\text{SEL}} \cup N^{\text{SEL}}$
16: $\quad$**end for**
17: $\quad$**return** S
18: **end procedure**

upper bound $\frac{S}{|\mathcal{C}|}$ determined by the backbone compression (S) and the number of object categories, while the latter focuses on ignoring a sub–portion of the backbone and, consequently, one or more features. In real scenarios involving deep learning–based detectors, limiting the number of heads as suggested by Corollary 1 to balance both detection and privacy is a promising solution as proposal–based detectors use a large number of heads densely distributed across the input [43]. On the contrary, the approach outlined in Corollary 2 is impractical as it prevents the detector from finding objects in those input areas that lack proposal coverage.

A. Co–training with Weak Loss

Following these intuitions, we propose a co–training scheme to obtain both detection and privacy. It consists of training an encoder–decoder optimized by a "weakened" task loss calculated using a (very) limited but meaningful subset of proposals. As suggested by Corollary 1, this approach forces the encoder–decoder to extract from the input x only the essential features to activate a small but targeted subset of TaskNet proposals, while discarding those features exploitable by potential attackers for reconstructing the original x from its obfuscated version $\tilde{x}$.

Our method is detailed in Algorithm 1. First, the encoder–decoder parameters are randomly initialized (line 1). Then for a given number of iterations, it randomly samples an example (x, Y) from the training set $\mathcal{D}$ (line 3), where x is the image and Y the set of ground truth bounding boxes. The next step aims at computing the set of unfiltered bounding boxes $\tilde{Y}$, which are computed by the TaskNet using the obfuscated version of the image x (line 4). The encoder–decoder parameters are updated by backpropagating the same loss used for training TaskNet (line 6), but computed over a reduced set of selected bounding–box proposals (line 5). The TaskNet's loss is thus "weakened" with respect to its original counterpart, which integrated all the large and dense set of proposals. The TaskNet leverages thousands of proposals with high overlap and low confidence from which the most promising are selected using heuristic algorithms like Non–Maximum Suppression and thresholding [43]. Proposal selection (defined from line 8) operates on a single example, extracting two types of proposals: *negative* and *positive*. The procedure, takes as arguments the ground truth bounding boxes (Y_{GT}), the dense proposal set computed by TaskNet (Y_t), and the number of required positive (p) and negative (n) proposals. Call $\rho(t, g)$ the intersection–over–union area (IoU) between a TaskNet's proposal t and a ground–truth bounding box $g \in Y_{GT}$. Also, denote with $\sigma(t)$ the confidence that TaskNet assigns to proposal t. A proposal t matches $g \in Y_{GT}$ if $\rho(t, g)$ is larger than that with any other $g' \neq g$. For $g \in Y_{GT}$, we define the positive proposals as those matching with g (line 11) and we select the first p in terms of IoU with g (line 12). Negative proposals, instead, are obtained with the same matching rule, but after ruling out the positive ones and imposing an upper bound $\bar{\rho}$ on the IoU (line 13). The algorithm selects the first n in terms of confidence (line 14). The rationale is that positive and negative proposals should configure as a "sparse", yet task–relevant, set on which evaluating the loss of a GT g. Positive proposals represent likely correct predictions, these are essential to learn the task. Negative proposals represent likely wrong ones, falling outside of the object region with high confidence. Note that our method can be run offline in combination with any off–the–shelf object detector working with dense proposals, such as YOLO [45], since it modifies only the number of bounding boxes used by the loss function.

Fig. 3 reports a visual example of the proposal selection procedure in Algorithm 1 with $p = n = 1$ and (IoU threshold) $\bar{\rho} = 0.5$. At first, an image x is processed by the EDO and then fed to TaskNet which produces a dense set of proposals $\tilde{Y}$. Fig. 3a depicts the ground truths (g_A and g_B) of x and some relevant proposals from $\tilde{Y}$. Then, our algorithm matches each proposal $\tilde{y} \in \tilde{Y}$ with the ground truth g that obtains the largest IoU area $\rho(\tilde{y}, g)$ with respect to all the others $g' \neq g$ (line 11). This can be seen in Fig. 3b, where the proposals matched with g_A (g_B) are depicted with a continuous (dashed) line. After matching, our algorithm selects, for each ground truth g, the best p proposal in terms of IoU area with their corresponding g (line 12), that are then removed from further computation (line 13). In our example, y_1 and y_4 are selected as positive proposals for g_A and g_B, respectively (see Fig. 3b). From the remaining bounding boxes, our algorithm defines the negative proposals for each g (shown in Fig. 3c) using the same matching rule but considering only those with a poor overlap with g, specifically $\rho(\tilde{y}, g) < \bar{\rho}$ (line 13). For example, $\tilde{y}_2$ is discarded as its IoU area with g_A exceeds $\bar{\rho}$. Note that the proposals marked as positive in the previous step are not considered in this phase, even if their intersection with a ground truth is below the threshold (like y_4 for which $\rho(\tilde{y}_4, g_B) < \bar{\rho}$). The computation proceeds by choosing the most confident n negative predictions for each ground truth

IEEE Transactions on Robotics (T-RO) paper, presented at ICRA 2026, Vienna, Austria. Cite as T-RO paper.

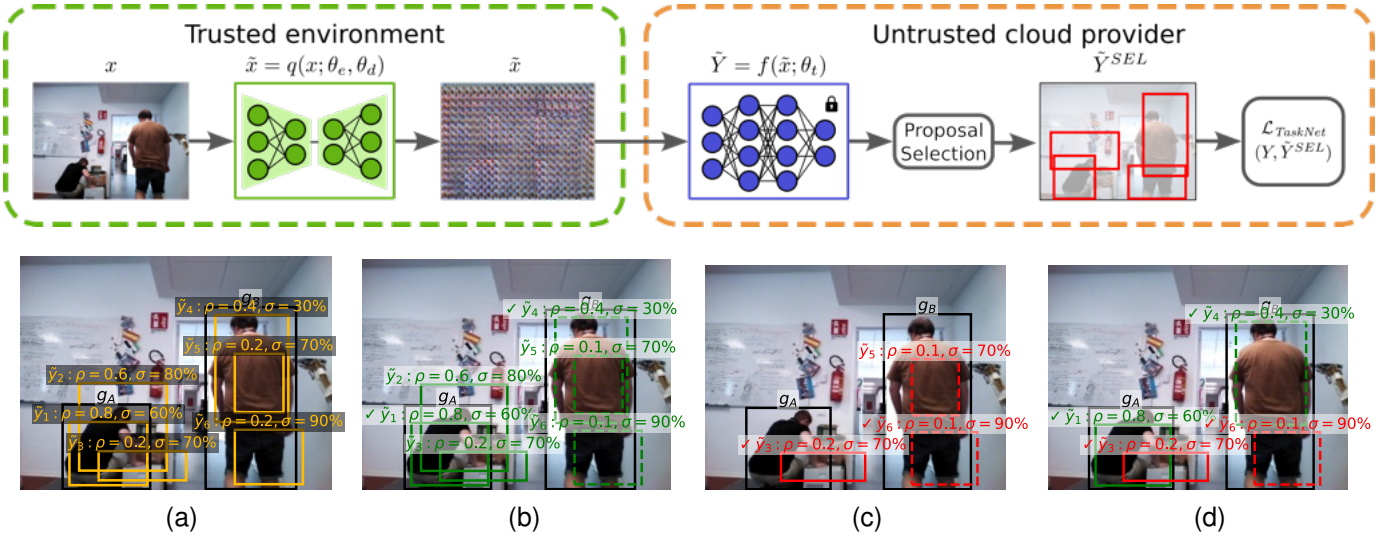


Fig. 3. Our co–training scheme for detection and privacy detailed in Algorithm 1. (First row) Given an image x and the ground truth bounding boxes Y, the encoder–decoder obfuscator is trained using the loss function of the TaskNet (whose weights are frozen) calculated between Y and a subset ($\tilde{Y}^{SEL}$) of the proposals produced by the TaskNet on obfuscated images $\tilde{x}$. (Second row) An example of the proposal selection procedure of Algorithm 1 applied on a perception acquired by our Giraff [24]. (a) The GT bounding boxes g_A and g_B (in black) and some proposals from the TaskNet's dense set (in orange). Note that, for visualization purposes, we depict only a few proposals among the thousands produced by the TaskNet, many of which overlap and have low confidence and IoU. (b) Positive proposals matched with g_A (in green) and g_B (in dashed green); the selected proposals are marked with a ✓. (c) Negative proposals associated with g_A (in red) and g_B (in dashed red) with $\bar{\rho} = 0.5$. (d) The final $\tilde{Y}^{\text{SEL}}$ when $p = n = 1$.

(line 14). Fig. 3d shows the final set of selected proposals $\tilde{Y}^{\text{SEL}} = \{\tilde{y}_1, \tilde{y}_3, \tilde{y}_4, \tilde{y}_6\}$.

V. Experimental Evaluation

A. Experimental Setting

The primary evaluation of our approach focuses on a highly privacy–sensitive object–detection task concerning service robots, specifically *people detection*. We implemented the related TaskNet with a standard approach that a third–party cloud provider can use: fine–tuning a stable off–the–shelf object detector using a publicly–available dataset for people detection. Following this, we use Faster R–CNN [44] (composed by ResNet–50 [48] as backbone) which is then fine–tuned using the (plain) ≈ 64k images from COCO 2017 dataset [50] that contain the object category `person`. Note that any proposal–based OD method could be used instead.

The encoder–decoder obfuscator (EDO) is implemented with 4 convolutional and de–convolutional blocks, that are composed of a pair of convolution layers (with 3 as kernel size and ReLU as activation function) followed by max pooling (for the former) and upscaling (for the latter). To simulate the scenario considered in this paper, where service robots need to obfuscate their visual perceptions for remote inference, we train our EDO using only a subset of the COCO images containing `person`, specifically those collected indoors. To extract this subset, we select images with at least a `person` along with another object category commonly found indoors. Examples are `table` or `refrigerator`; the full list is reported in Table II. Note that this sample selection procedure maintains the proportions between the train and validation splits of COCO, reducing them from 64k to 11k and from 2.5k to 500 images, respectively. Training is carried out using our co–training scheme with weak loss of Algorithm 1 for different amounts of positive p and negative n proposals; specifically, we consider $p \in \{1, 2, 3, 4\}$ and $n \in \{0, 1, 2, 3, 4\}$.

TABLE II
Categories from COCO 2017 [50] used to identify images likely acquired in indoor environments. The images labeled with these categories have been used to train our EDO architecture.

bench	bird	cat
dog	backpack	umbrella
handbag	tie	suitcase
bottle	wine glass	cup
fork	knife	spoon
bowl	banana	apple
sandwich	orange	broccoli
carrot	hot dog	pizza
donut	cake	chair
couch	potted plant	bed
dining table	toilet	tv
laptop	mouse	remote keyboard
cell phone	microwave oven	toaster
sink	refrigerator	book
clock	vase	scissors
teddy bear	hair drier	toothbrush

The attack model of Problem 2 is implemented using the same architecture as EDO. The attacker is tasked to reconstruct the original images from those that have been obfuscated with our method. We follow the Model Inversion Attack (MIA) approach described in [41]; thus, training is performed with Mean Absolute Error (MAE), that is also the natural implementation of the general–purpose reconstruction loss defined in our theoretical framework (see Theorem 2). We used the same hyperparameters of the EDO, but we increased the number of epochs (from 50 to 80) to ensure the best attacker–reconstruction performance. Note that we train, for each EDO, a different attacker using its obfuscated images obtained from the same training dataset, thus relying on the most powerful

IEEE Transactions on Robotics (T-RO) paper, presented at ICRA 2026, Vienna, Austria. Cite as T-RO paper.

MIA according to [23], [41].

To further assess the privacy–preserving capabilities of our method, we introduced an enhanced attacker model trained with the edge–centric (EC) loss function from [41], which improves reconstruction power by considering not only individual pixel differences (as in MAE), but also local pixel surroundings. Following the insights of [41], this enhanced loss function is strictly related to privacy as it promotes the reconstruction of sensitive fine–grained details, such as facial features or small textures. The final loss function of this improved MIA is defined as:

$$\|x - \hat{x}\|_1 + \beta\|S_h * x - S_h * \hat{x}\|_1 + \beta\|S_v * x - S_v * \hat{x}\|_1, \quad (3)$$

where S_h and S_v are the horizontal and vertical Sobel kernels while $(*)$ is the convolution operator. As in [41], β is set equal to 5. We test this configuration with $p = n \in \{1, 2, 3, 4\}$ and we report the results in Section V-D.

We compare our approach for privacy preservation with a state–of–the–art baseline derived from [22], where we replaced, following the considerations detailed in the previous section, the VAE with the larger EDO trained with Algorithm 1 without performing proposal selection, so using all the TaskNet's proposals (label ALL). As a reference, we report the upper bound of the performance achieved by the TaskNet on plain images (label TN). Note how, differently from the assumptions made in our reference scenario, in the TN setup the cloud provider must be trusted.

Testing is performed on the COCO validation split, filtered for indoor examples ($\approx$ 500 images) as described above. Furthermore, we validate our approach in out–of–distribution settings using the validation split of Pascal VOC 2012 [51] dataset ($\approx$ 2k images acquired indoors and outdoors). Differently from testing on COCO, we here do not perform the filtering procedure in order to further challenge and assess the robustness of our method.

To assess generalizability across various privacy–demanding tasks, we conduct an additional evaluation campaign focused on the multi–class task of *vehicle detection*. Specifically, we run the experiments described above with `bicycle`, `airplane`, `bus`, and `train` as object categories. We train our framework with the COCO dataset following the same procedure used for people detection but considering only two proposal configurations, $p = n \in \{2, 3\}$. Also in this case, the performances are evaluated on the validation splits of COCO and the out–of–distribution instances from Pascal VOC.

The full details on the hyperparameters adopted for training the aforementioned modules (TaskNet, EDO, and the attackers) in the two considered tasks are reported in Table III. Additionally, we enrich training with data augmentation by implementing random horizontal flip (with a probability of 0.5) and random resize in which the images are rescaled (maintaining the aspect ratio) setting the length of the smallest dimension to each of the values in $\{256, 288, 320, 352, 384, 416\}$. Our implementation is based on PyTorch and, since the code makes use of random calls in different steps (for instance in line 1 of Algorithm 1), we ensure full reproducibility by fixing the random seeds in every random call, notably in those performed by `torch`, `os`, `numpy`, and `random` libraries. Our code, along with all other low–level implementation details, is made accessible in a publicly available repository[2].

TABLE III
HYPERPARAMETERS USED TO TRAIN TASKNET, EDO, AND THE ATTACKER'S ENCODER–DECODER IN THE EXPERIMENTS REPORTED.

	People detection			Vehicle detection		
	TaskNet	EDO	Attacker	TaskNet	EDO	Attacker
Epochs	10	50	80	10	50	80
Batch Size	8	4	4	2	2	2
Optimizer	SGD	SGD	SGD	SGD	SGD	SGD
Learning Rate	1e-3	5e-4	5e-4	4e-4	3e-4	3e-4
Weight Decay	5e-4	5e-4	5e-4	5e-4	5e-4	5e-4
Nesterov Momentum	0.9	0.9	0.9	0.9	0.9	0.9
Scheduler	StepLR	RLROnP	RLROnP	StepLR	RLROnP	RLROnP
Step Size / Gamma	3 / 0.1	–	–	3 / 0.1	–	–
Patience / Factor	–	2 / 0.5	4 / 0.5	–	2 / 0.5	4 / 0.5

We complement the above empirical evaluation by testing our method on real robotic platforms (see Section V-E). To achieve this, we deploy our approach using Giraff, a service autonomous robot that features a camera with a resolution of 256×256 pixels. This platform, depicted in Fig. 1 and detailed in [24], has been widely used in human–centric assisted living environments where addressing privacy concerns is crucial but largely neglected. We gathered a stream of images as the robot autonomously navigated through our university building, where people moved and walked by freely. We sampled the robot's perceptions at 1 Hz, resulting in approximately 250 examples, and we annotated all instances of people appearing in the images. This dataset was exclusively used for testing purposes. Additionally, Sec. V-F evaluates the computational demands of our implementation by measuring the inference time required on low–powered hardware typically utilized in mobile robot configurations. We carried out assessments on Giraff, which is fitted with an NVIDIA Jetson TX2 (GPU), and also conducted tests using a TurtleBot3 equipped with a Raspberry PI 4 (CPU).

B. Performance Metrics

The evaluation of performance in both tasks is conducted using the standard Average Precision (AP) and AP_{50} metrics from COCO [50]. The AP and AP_{50} are customary metrics that offer a reliable assessment of object detection performance across varying parameters.

Apart from these AP–based metrics (commonly used in object detection), we incorporate two additional performance metrics that are more relevant to the deployment of the object detection module in our robotic setting and that we proposed in [52]: True Positive (TP) and Background False Detection (BFD) rates. True positive (TP) is defined as the rate at which ground truth bounding boxes are accurately paired with at least one prediction. A pair is considered a match when both the predicted and ground truth (GT) labels are the same and their Intersection over Union (IoU) area is greater than a given threshold ρ_{IoU}. On the other hand, Background False Detection is calculated as the proportion of predictions, normalized by the total GTs, that end up in the background (i.e., having an IoU area with all GTs below the threshold ρ_{IoU}). In this analysis, we focus only on the predictions with

[2]https://aislab.di.unimi.it/research/privacyweakloss

IEEE Transactions on Robotics (T-RO) paper, presented at ICRA 2026, Vienna, Austria. Cite as T-RO paper.

TABLE IV

AP, AP_{50}, AND MS–SSIM (MS) RESULTS FOR DIFFERENT VALUES OF p AND n, REPRESENTING THE NUMBER OF POSITIVE AND NEGATIVE PROPOSALS SELECTED DURING CO–TRAINING. RESULTS ARE COMPARED AGAINST TWO BASELINES: TRAINING WITH ALL PROPOSALS (ALL) AND USING THE TASKNET ON PLAIN (NON–OBFUSCATED) IMAGES (TN). FOR TN, MS–SSIM IS SET TO 100, INDICATING NO PRIVACY PROTECTION. THE RESULTS HIGHLIGHT THE TRADE–OFF BETWEEN TASK PERFORMANCE AND PRIVACY: INCREASING p AND n GENERALLY ENHANCES DETECTION ACCURACY (HIGHER AP AND AP_{50}), BUT REDUCES PRIVACY (HIGHER MS). INTERMEDIATE SETTINGS (E.G., $p = 2$, $n = 2$) OFFER A FAVORABLE BALANCE BETWEEN THE TWO OBJECTIVES.

	COCO [50]												Pascal VOC 2012 [51]											
	AP ↑				AP_{50} ↑				MS ↓				AP ↑				AP_{50} ↑				MS ↓			
$p =$	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
$n = 0$	18	30	33	35	40	57	60	62	36	45	46	47	14	25	27	28	34	51	54	56	35	44	45	46
$n = 1$	26	33	36	32	52	60	64	61	46	51	46	53	20	27	30	27	44	54	58	53	44	50	45	52
$n = 2$	32	36	40	41	59	64	68	69	43	46	54	57	25	28	32	32	50	56	61	60	42	45	53	55
$n = 3$	30	32	38	39	58	60	67	68	44	46	53	51	24	26	31	31	51	53	59	59	43	44	52	49
$n = 4$	29	34	34	40	56	61	62	69	45	46	46	53	23	27	28	33	48	54	55	60	44	45	45	52
ALL	47				76				69				39				68				68			
TN	59				87				100				55				86				100			

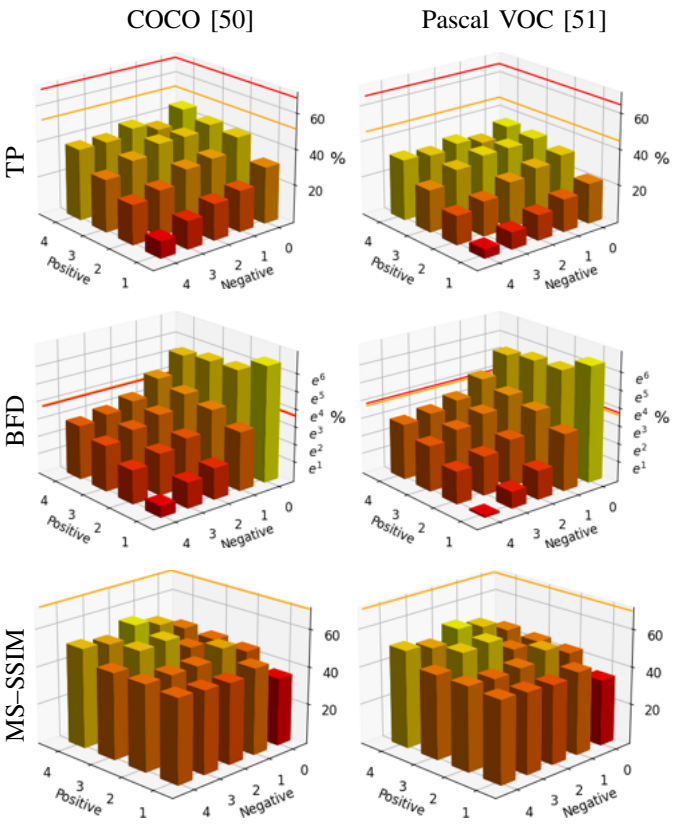


Fig. 4. True Positive rate (TP), Background False Detection rate (BFD), and MS–SSIM (MS) for our co–training scheme under different values of p and n (positive and negative proposals). The orange line represents the performance of the EDO trained with all proposals (ALL), while the red line denotes the TaskNet operating on plain images (TN). For reference, TN's MS–SSIM is fixed at 100, indicating no privacy. The results further illustrate the trade–off between perception performance and privacy.

the highest confidence, selecting those predictions for which the probability of the predicted object category exceeds a threshold ρ_c. To maintain a conservative assessment, we have set $\rho_{IoU} = \rho_c = 75\%$.

The potential privacy breach by the attacker is assessed utilizing the Multi–Scale Structural Similarity Index Measure (MS–SSIM) as defined in [53], with reconstructed images $\hat{x}$ serving as inputs. This metric evaluates the perceptual quality of an image after undergoing a degradation process. It is widely recognized as a method to quantify the utility of an altered image, using information degradation as a proxy. The score ranges from 100, representing a perfect match to the original image, to values approaching 0, indicating a significant loss of structural and perceptual information with respect to the original. In the following tables and charts we report averaged values obtained across the images test sets. Similarly, L–PIPS [54], another well–recognized metric, was evaluated alongside MS–SSIM. As both metrics yield largely consistent findings, the results with L–PIPS are omitted for brevity.

C. Method Evaluation

The results reported in Table IV show that our co–training scheme based on weak loss induces a tunable trade–off between perception performance and privacy, enabling the encoder–decoder to discard sensitive information, reducing the reconstruction power of the attacker while preserving the necessary features to allow object detection. Specifically, the detection capability (AP and AP_{50}) of the TaskNet on obfuscated images increases with higher values of p and n while privacy improves by reducing the amount of proposals used in training. It is interesting to observe how using (very) few proposals for training the EDO produces a marginal decrease in performance while drastically increasing privacy with respect to the ALL baseline (see Fig. 5), where thousands of proposals are used. As an example, on the COCO dataset, when $p = n = 4$, AP_{50} has a $\approx 9\%$ drop against ALL while MS–SSIM improves by $\approx 23\%$. This outcome is further corroborated by the images from the Pascal VOC dataset, validating the robustness of our method in out–of–distribution settings. In particular, with Pascal VOC's data, AP_{50} decreases by $\approx 12\%$ while MS–SSIM improves by $\approx 24\%$ when $p = n = 4$ with respect to the ALL baseline.

From the additional indicators, whose values are reported in Fig. 4, we can highlight and examine a performance trade–off not captured by the standard AP. On the one hand, increasing the positive proposals p allows the TaskNet to reach a better

IEEE Transactions on Robotics (T-RO) paper, presented at ICRA 2026, Vienna, Austria. Cite as T-RO paper.

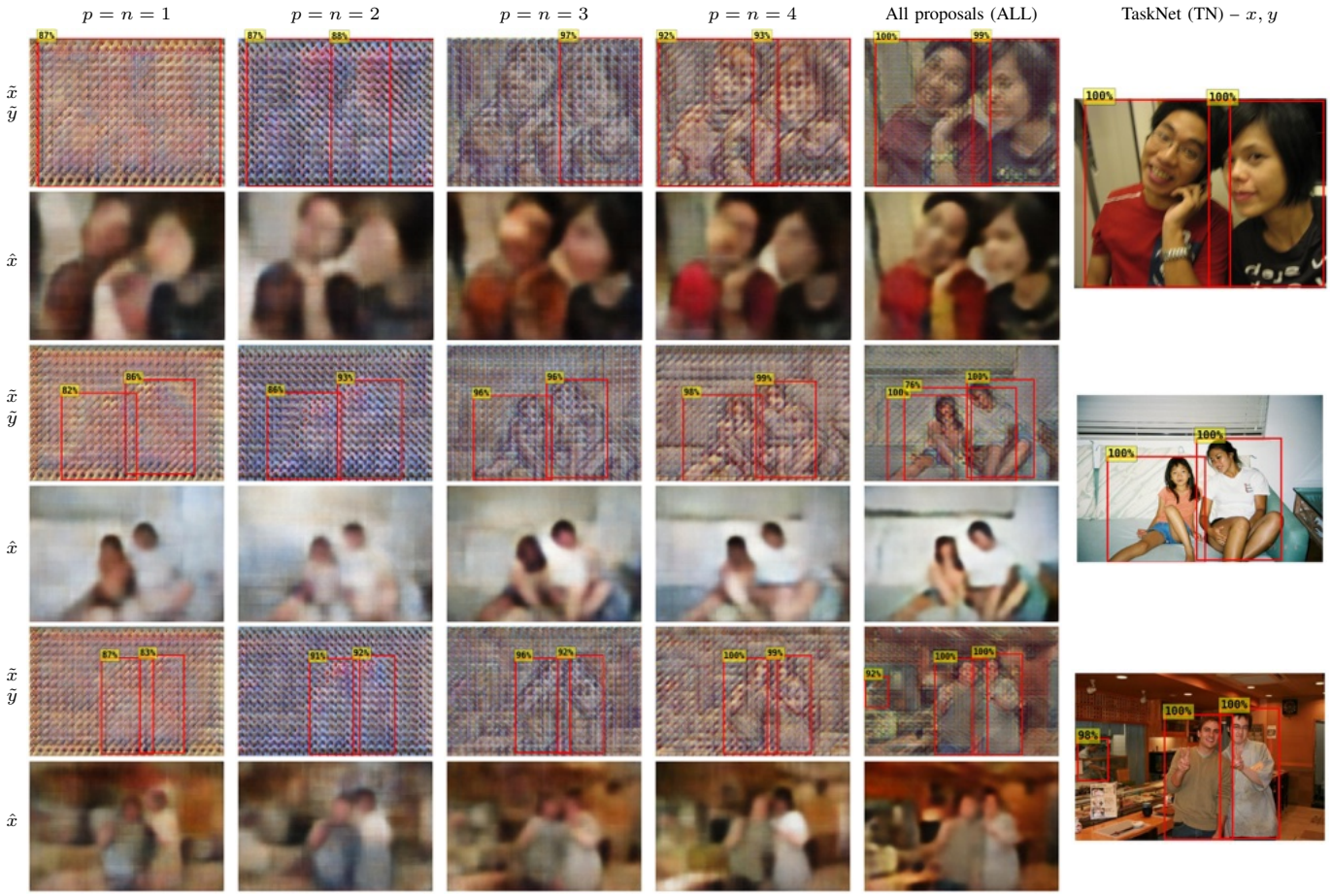


Fig. 5. Examples of our method on out–of–distribution images from Pascal VOC 2012 [51]. (Left) Obfuscated images ($\tilde{x}$) and their reconstructions ($\hat{x}$) generated by the attacker. (Right) Corresponding plain images (x) with TaskNet detections (y) shown for reference. Red bounding boxes indicate TaskNet predictions (y and $\tilde{y}$) with confidence scores $\sigma(y), \sigma(\tilde{y}) \geq 0.75$, filtered using Non–Maximum Suppression (NMS) with an IoU threshold of 0.5. Results show that, despite strong obfuscation, TaskNet retains detection capabilities on $\tilde{x}$, while reconstruction quality remains low.

true positive rate (TP), obtaining values close to the ALL baseline both with COCO and Pascal VOC datasets. In particular, when setting $p = n = 4$, there is a reduction of $\approx 19\%$ and $\approx 21\%$ in true positives (TPs) compared to the ALL baseline for COCO and Pascal VOC images, respectively. On the other hand, increasing the negative proposals n reduces the number of false positive bounding boxes on the background (BFD). As expected, training the EDO without considering negative proposals ($n = 0$) results in obfuscated images that fail to suppress the false positive predictions produced by the TaskNet. This can be seen in the second row of Fig. 4, which shows a significantly higher BFD rate compared to the ALL baseline. Interestingly, the BFD values obtained by the TaskNet on obfuscated images using 2 or more negative proposals are remarkably lower than those obtained both with the ALL baseline and the TaskNet on plain images (TN). This demonstrates that our approach reduces the number of errors, making the detection process more conservative. In particular, training the EDO using only 4 positive and negative proposals drops the BFD of $\approx 36\%$ and $\approx 29\%$ compared to using all proposals on COCO and Pascal VOC, respectively.

Overall, our extensive experimental campaign gives some guidelines for choosing the values of p and n to balance detection and privacy. Higher positives (p) increase the true positive predictions (TP) while the errors (BFD) are reduced by increasing the negatives (n). These combined findings are due to the fact that, while the p proposals promote the activation of bounding boxes close to the targets, the n ones force the suppression of spurious detections (BFD). Given this, the plots in Fig. 4 suggest that choosing values of p and n close to each other ensures a good compromise between TP and BFD, in particular when $p = n \in \{2, 4\}$. This can be seen by comparing the raw values reported in Table IV: while the configurations with $p = n \in \{1, 2\}$ and $p = n \in \{3, 4\}$ reach comparable obfuscation results, setting $p = n = 4$ ($p = n = 2$) increases the detection performance compared to $p = n = 3$ ($p = n = 1$).

The benefit of reducing the number of proposals can be qualitatively appreciated in the out–of–distribution examples of Pascal VOC reported in Fig. 5. At first, we can notice how the images obfuscated with the baseline (ALL) are similar to the original inputs and preserve privacy–sensitive details that enable the attacker to obtain a good reconstruction of the original input. In particular, the baseline retains characteristics like facial expressions, hairstyles, body silhouettes, and background elements, which not only help identify individuals but

IEEE Transactions on Robotics (T-RO) paper, presented at ICRA 2026, Vienna, Austria. Cite as T-RO paper.

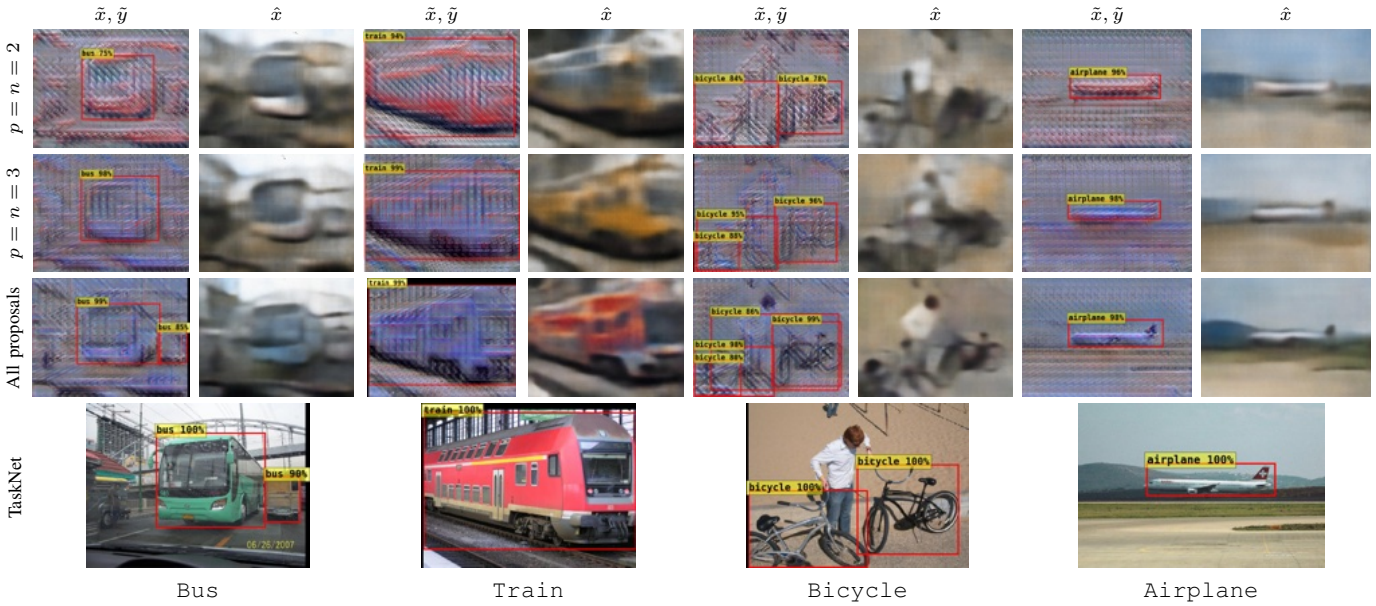


Fig. 6. Qualitative examples for the vehicle detection task on out–of–distribution images from Pascal VOC 2012 [51]. (Top) Obfuscated images ($\tilde{x}$) with corresponding TaskNet detections ($\tilde{y}$) and reconstructions by the attacker ($\hat{x}$). (Bottom) Plain images (x) and TaskNet detections (y) used as reference. Red bounding boxes indicate TaskNet predictions (y and $\tilde{y}$) with confidence scores $\sigma(y), \sigma(\tilde{y}) \geq 0.75$, filtered using Non–Maximum Suppression (NMS) with an IoU threshold of 0.5. These examples illustrate that our method preserves detection capabilities on obfuscated images also in a multi-class detection task.

also offer context regarding the locations where the images were taken. Conversely, the privacy ensured by our method, with both p and n set to 4, is remarkably better than those obtained with the ALL baseline. Furthermore, reducing the number of proposals further enhances the level of obfuscation, removing critical visual clues such as clothing colors and altering the shapes of faces, bodies, and objects in the scene, preserving user identity. Despite the obfuscation provided by the EDO, the TaskNet maintains robust detection performance even on privatized images.

These findings are confirmed also in the vehicle detection task. The results reported in Table V confirm that our method strongly improves privacy (lower MS–SSIM compared to ALL) while preserving good detection performance. In particular, setting $p = n = 3$ reduces the AP of $\approx 11\%$ while MS–SSIM improves by $\approx 19\%$ on COCO. This trade–off is also more evident in the Pascal VOC benchmark, where the AP decreases by $\approx 5\%$ while MS–SSIM shows a remarkable improvement of $\approx 20\%$. Interestingly, the performances obtained with the out–of–distribution dataset of Pascal VOC are better than those obtained using the in–distribution COCO. This is due to the fact that COCO images contain some challenging small targets that are difficult to detect in obfuscated images. In the task of people detection, using 2 or 3 positive and negative proposals enhances MS–SSIM, maintaining similar detection performance. Conversely, for vehicle detection, setting $p = n = 2$ reduces detection performance without offering any privacy advantage over $p = n = 3$. This indicates that, as seems reasonable, the number of proposals in detection tasks involving multiple object categories should be marginally higher than in single–class object detection.

Visual examples of out–of–distribution data from Pascal VOC can be seen in Fig. 6. Also in this context, the ALL baseline preserves critical features from the original images, such as the shape and color of vehicles, as well as other contextual details in the background, allowing the attacker to restore privacy–sensitive information. In contrast, our method significantly enhances privacy, making it extremely difficult to distinguish the types of vehicles. For instance, the outline of the bus strongly degrades, the color and shape of the train are completely lost, the person near the bicycles is mixed with the background, and the features to identify the airplane (such as the tail and wings) completely disappear. Despite this, our method is still able to perform vehicle detection also with the challenging obfuscated images. Interestingly, our method solves some errors produced by the ALL baseline, such as the small van near the bus or the multiple overlapped bicycles.

TABLE V
PERFORMANCE OF OUR METHOD IN A 4–CLASS VEHICLE DETECTION SCENARIO. RESULTS ARE REPORTED FOR TWO SETTINGS OF THE WEAK LOSS PARAMETERS ($p = n = 2$ AND $p = n = 3$), AND COMPARED WITH THE TASKNET OPERATING ON PLAIN IMAGES (TN). THE RESULTS SHOW THAT OUR APPROACH GENERALIZES TO MULTI-CLASS SETTINGS.

	COCO [50]					Pascal VOC 2012 [51]				
p, n	AP↑	AP_{50}↑	TP↑	BFD↓	MS↓	AP↑	AP_{50}↑	TP↑	BFD↓	MS↓
2, 2	23	38	26	42	46	33	54	38	42	49
3, 3	30	48	35	45	47	42	64	49	56	49
ALL	34	56	38	40	58	44	71	51	50	61
TN	52	77	59	33	100	63	88	74	30	100

D. Evaluation With an Enhanced Model Inversion Attack

We further evaluate the protection provided by our method against MIA threats. To do this, we enhance the reconstruction power of the attacker using the loss function of Eq. 3 that aims to recover the privacy–sensitive features from the obfuscated

IEEE Transactions on Robotics (T-RO) paper, presented at ICRA 2026, Vienna, Austria. Cite as T-RO paper.

TABLE VI
COMPARISON OF THE RECONSTRUCTION POWER (MEASURED WITH MS–SSIM) OF THE MIAS TRAINED WITH DIFFERENT LOSS FUNCTIONS: THE MEAN ABSOLUTE ERROR (MAE) AND THE ENHANCED EDGE–CENTRIC LOSS OF EQ. 3 (EC). RESULTS SHOW HOW THE ATTACKER'S RECONSTRUCTION POWER DOES NOT SUBSTANTIALLY CHANGE FOR A MORE SOPHISTICATED ATTACKER.

p, n	COCO MAE	COCO EC	Pascal VOC MAE	Pascal VOC EC
$1, 1$	46	45	44	43
$2, 2$	46	46	45	45
$3, 3$	53	54	52	53
$4, 4$	53	53	52	51
ALL	69	69	68	68

image. Table VI report the MS–SSIM performance achieved by the two different attacker models on both in– and out–of–distribution datasets (COCO and Pascal VOC) with the EDO trained with different proposal configurations.

The results demonstrate that our method is robust to enhanced attackers using more sophisticated loss functions. The MIA trained with the EC loss reaches MS–SSIM values (very) close to the MIA using the simpler MAE. This further corroborates the effectiveness of our method in compromising the reconstruction power of a malicious actor by removing privacy–sensitive features that are redundant for the task execution.

Fig. 7 shows qualitative examples comparing the reconstruction obtained by the MAE and EC attackers. Our method is robust against both the simpler MAE loss and the more complex EC one: the reconstructed images obtained by the two attack models are similar at first glance, and only a closer inspection reveals the few positive effects brought by the more powerful EC loss. As an example, the images reconstructed by the attacker using the EC loss have less noise in uniform areas (e.g., walls), and have slightly better details in edges and contours. Still, data obfuscated by our method and reconstructed with the more powerful EC loss contain significant less details when compared against data reconstructed from images obfuscated with our baseline method (EDO trained with all proposals). While the enhanced loss indeed produces minor improvements from the point of view of the attacker when compared with the less powerful MAE, it fails to restore privacy–sensitive fine details, thus demonstrating again the effectiveness of our approach.

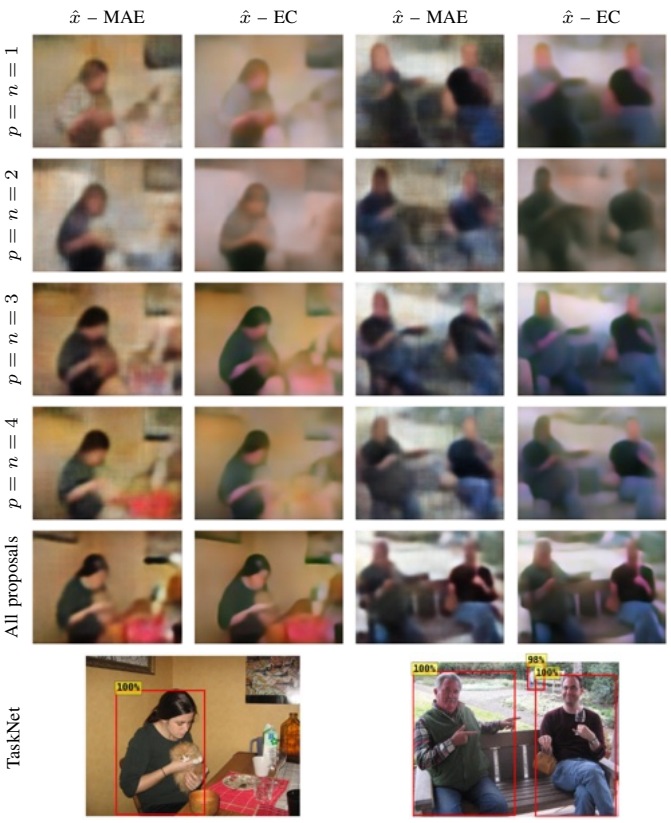


Fig. 7. Qualitative comparison of two MIA strategies trained with different loss functions: Mean Absolute Error (MAE) and the enhanced edge–centric loss (EC) defined in Eq. 3. Reconstructions ($\hat{x}$) are shown for various EDO configurations on out–of–distribution examples from Pascal VOC 2012 [51]. (Top) Reconstructed images produced by the attacker. (Bottom) Corresponding plain images (x) and TaskNet detections (y) shown for reference. Red bounding boxes indicate TaskNet predictions with confidence $\sigma(y) \geq 0.75$, filtered using Non–Maximum Suppression (NMS) at IoU threshold 0.5. Despite the more sophisticated EC loss, visual inspection reveals no substantial differences.

E. Real–Robot Evaluation

In this section, we assess the efficacy of our approach when deployed on a real mobile robot, which has to autonomously carry out the task of people detection while navigating in an indoor environment. Motivated by the results reported in Section V-C, we test our framework setting $p = n \in \{1, 2, 3, 4\}$. The results presented in Table VII, obtained with Giraff, show that our approach achieves satisfying detection and privacy preservation capabilities. Our method remarkably improves obfuscation while maintaining detection capabilities very close to the ALL baseline. Specifically, setting $p = n = 4$ degrades AP_{50} from 87 to 85 ($\approx 2\%$) while improving MS–SSIM from 81 to 61 ($\approx 25\%$). Again, reducing p and n slightly degrades detection but strongly increases privacy. As an example, setting $p = n = 2$ further improves privacy by $\approx 33\%$ (MS–SSIM) at the cost of $\approx 4\%$ AP_{50} drop. Another interesting fact can be observed by comparing the AP, TP, and BFD performance of the configurations where $p = n \in \{2, 3, 4\}$. While the AP values remain almost the same, the true positive (TP) and spurious detection in the background (BFD) report an evident decreasing trend. Specifically, when the number of proposals is reduced from 4 to 2, TaskNet misses $\approx 16\%$ of the ground truths (with TP decreasing from 80 to 67) but, at the same time, the rate of error is halved (BFD drops from 11 to 5). This further demonstrates that our method makes the detection process more conservative as it reduces the number of false positive detections while maintaining the AP stable. Similar to the people detection evaluation on COCO, setting $p = n \in \{2, 4\}$ better balances detection and privacy compared to $p = n \in \{1, 3\}$: using 4 (or 2) positive and negative proposals ensures a comparable level of privacy with higher detection performance than using 3 (or 1) proposals per type.

Fig. 8 visualizes some representative examples of this evaluation. The images have been acquired from the point of

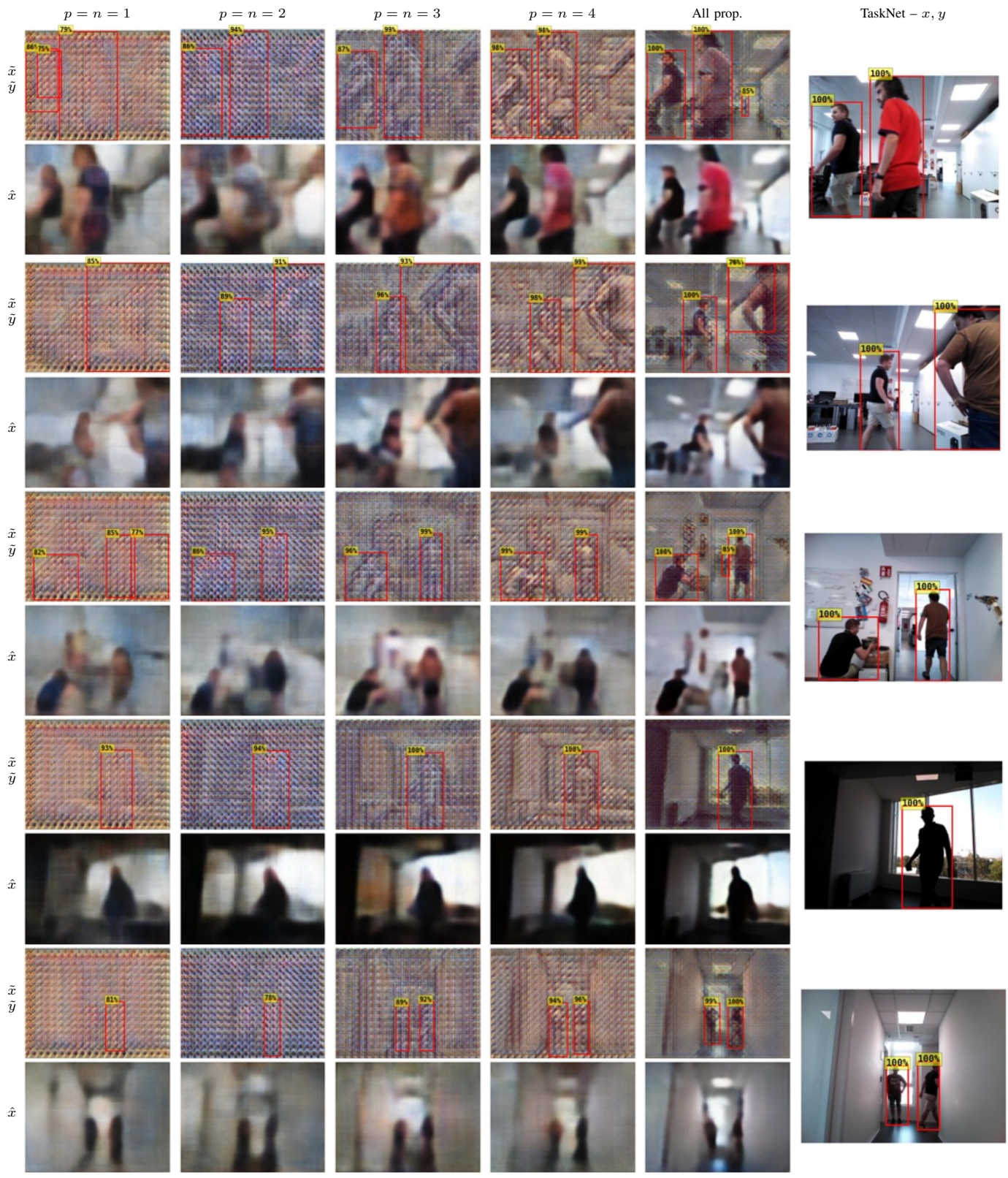


Fig. 8. Qualitative examples of our method applied to images acquired from the onboard camera of the Giraff robot during real–world operation. (Left) Obfuscated images ($\tilde{x}$) and corresponding reconstructions ($\hat{x}$) generated by the attacker. (Right) Plain input images (x) with TaskNet detections (y) shown for reference. Red bounding boxes indicate TaskNet predictions (y and $\tilde{y}$) with confidence scores $\sigma(y), \sigma(\tilde{y}) \geq 0.75$, filtered using Non–Maximum Suppression (NMS) at an IoU threshold of 0.5. The examples highlight the ability of our method to preserve detection performance while ensuring privacy in real robotic deployments.

IEEE Transactions on Robotics (T-RO) paper, presented at ICRA 2026, Vienna, Austria. Cite as T-RO paper.

TABLE VII
DETECTION AND PRIVACY PERFORMANCE OF OUR METHOD DEPLOYED ON THE GIRAFF ROBOTIC PLATFORM FOR THE TASK OF PEOPLE DETECTION. RESULTS ARE REPORTED IN TERMS OF AP, AP_{50}, AND MS–SSIM (MS). THESE RESULTS DEMONSTRATE THAT OUR METHOD REMAINS EFFECTIVE WHEN DEPLOYED ON A REAL ROBOT.

p, n	AP ↑	AP_{50} ↑	TP ↑	BFD ↓	MS ↓
$1, 1$	37	68	58	8	53
$2, 2$	56	83	67	5	54
$3, 3$	57	84	77	10	61
$4, 4$	59	85	80	11	61
ALL	66	87	86	17	81
TN	78	96	95	19	100

view of our robot freely moving in the environment (using its navigation stack) while performing people detection on obfuscated images (TaskNet detections are reported for reference). It is easy to see how the perceptions obfuscated by the ALL baseline have visual characteristics that closely resemble those of plain images. Consequently, the attacker can generate restored images that are similar to the original perceptions, albeit with a slight blurring effect. From Fig. 8, we can see how the ALL baseline preserves environmental features (such as the scene's structure or the furniture's outline) as well as visual clues related to the people allowing their identification. On the contrary, our method substantially increases data protection by removing important details from the privatized images. This is evident from the first two examples of Fig. 8 that contain people in the foreground (close to the robot's camera). Thanks to co–training with weak loss, the EDO strongly degrades the quality of the obfuscated images, preventing the attacker from reconstructing privacy–sensitive details such as the body shapes, the clothing colors, and the background structure, protecting people from identification. Moreover, the obfuscated images preserve the necessary features for detecting people with high precision, even in challenging instances such as those affected by poor illumination or when the targets are at a high distance from the robot (see, respectively, the last two examples of Fig. 8). A video of this experiment is available in the graphical abstract.

F. Computational Performance

To assess the computational requirements of our framework, we deploy the EDO on two real robotics platforms with different hardware configurations: a TurtleBot 3 equipped with a Raspberry PI 4 (CPU) and Giraff [24] mounting an NVIDIA Jetson TX2 (a GPU accelerator specifically designed for low–powered devices). On the two robots, EDO reaches 0.5 and 5 FPS in CPU and GPU, respectively, while processing a 256×256 camera stream. FPS can be further improved by reducing the dimensionality of the encoder–decoder obfuscator. We test this solution by halving the channels of the EDO's convolutional layers: this increases the FPS to 1 (16) in CPU (GPU) with a marginal loss in detection performance (≈ 2 AP points) while preserving MS–SSIM. To better contextualize the efficiency of our method, we test the framerate of the Faster R–CNN with a ResNet–50. It reaches 1 FPS on the Jetson TX2, while the Raspberry PI 4 didn't manage to process a single frame in a reasonable amount of time. Despite the GPU acceleration, the TaskNet is 5x and 16x slower than our EDO in the two configurations with full and halved channels. The TaskNet offers various configurations with different accuracy and size. On the robot's low–powered hardware, we used a TaskNet with a reduced number of parameters. This contrasts with the cloud provider, which can employ larger backbones (e.g., ResNet–152) to maximize detection accuracy [44]. Running these more extensive models directly on the robot is not feasible due to hardware limitations or further increases the inference time. In contrast, our EDO demonstrates superior efficiency and its dimension can be easily tuned according to the hardware setups. In addition, while the TaskNet saturates the computational capabilities of the device even at 1 FPS, our EDO performs inference in just 0.2 seconds (≈ 0.06 seconds in the halved configuration). This allows practitioners to limit the EDO's inference speed to run multiple inference tasks or to preserve energy, thus enhancing operational autonomy.

The frame rate achieved by our method is consistent with the latency required for online computation by the modern architectures for fog and cloud robotics, as the one reported in [55] and, in our specific task of object detection for robotic vision, the frame rate of our implementation is in line with that of [56]. To properly contextualize these results, consider that the output of object detection is often used for mission–critical decision tasks that do not require hard real–time performance.

VI. CONCLUSIONS

This work presents a novel approach to ensure privacy in cloud–based object detection for mobile robots. Specifically, we propose a co–training scheme with weak loss to build an encoder–decoder that removes sensitive information while maintaining task–relevant features from the robot's perceptions. In future work, we will validate our approach with other object detection methods and extend our method by adding a privacy loss function.

ACKNOLEDGEMENTS

This work was partially supported by PNRR, M4C2, Investment 1.3, PE00000013, "FAIR, Future Artificial Intelligence Research" funded by the European Commission under the NextGenerationEU programme.

REFERENCES

[1] G. Hu, W. P. Tay, and Y. Wen, "Cloud robotics: architecture, challenges and applications," *IEEE Network*, vol. 26, no. 3, pp. 21–28, 2012.

[2] M. Penmetcha, S. S. Kannan, and B.-C. Min, "Smart cloud: Scalable cloud robotic architecture for web-powered multi-robot applications," in *Proceedings of The International Conference on Systems, Man, and Cybernetics (SMC)*, 2020, pp. 2397–2402.

[3] M. Antonazzi, M. Luperto, N. A. Borghese, and N. Basilico, "R2snet: Scalable domain adaptation for object detection in cloud-based robotic ecosystems via proposal refinement," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024, pp. 2676–2682.

[4] W. J. Beksi, J. Spruth, and N. Papanikolopoulos, "Core: A cloud-based object recognition engine for robotics," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 4512–4517.

[5] F. Lumpp, F. Fummi, H. D. Patel, and N. Bombieri, "Enabling kubernetes orchestration of mixed-criticality software for autonomous mobile robots," *IEEE Transactions on Robotics*, vol. 40, pp. 540–553, 2024.

[6] L. Wen, Y. Zhang, M. Rickert, J. Lin, F. Pan, and A. Knoll, "Cloud-native fog robotics: Model-based deployment and evaluation of real-time applications," *IEEE Robotics and Automation Letters*, vol. 10, no. 1, pp. 398–405, 2025.

[7] J. Hsu, "Now you too can buy cloud-based deep learning," *IEEE Spectrum*, July 2021.

[8] M. B. Alatise and G. P. Hancke, "A review on challenges of autonomous mobile robot and sensor fusion methods," *IEEE Access*, vol. 8, pp. 39 830–39 846, 2020.

[9] M. Luperto, M. Romeo, J. Monroy, J. Renoux, A. Vuono, F.-A. Moreno, J. Gonzalez-Jimenez, N. Basilico, and N. A. Borghese, "User feedback and remote supervision for assisted living with mobile robots: A field study in long-term autonomy," *Robotics and Autonomous Systems*, vol. 155, p. 104170, 2022.

[10] D. J. Butler, J. Huang, F. Roesner, and M. Cakmak, "The privacy-utility tradeoff for remotely teleoperated robots," in *Proceedings of the tenth annual ACM/IEEE international conference on human-robot interaction*, 2015, pp. 27–34.

[11] A. K. Taras, N. Sünderhauf, P. Corke, and D. G. Dansereau, "Inherently privacy-preserving vision for trustworthy autonomous systems: Needs and solutions," *Journal of Responsible Technology*, vol. 17, p. 100079, 2024.

[12] E. Guo, "A roomba recorded a woman on the toilet. how did screenshots end up on facebook," *https://www.technologyreview.com/2022/12/19/1065306/roomba-irobot-robot-vacuums-artificial-intelligence-training-data-privacy*, 2022.

[13] Y. Guo, B. Zou, J. Ren, Q. Liu, D. Zhang, and Y. Zhang, "Distributed and efficient object detection via interactions among devices, edge, and cloud," *IEEE Transactions on Multimedia*, vol. 21, no. 11, pp. 2903–2915, 2019.

[14] C. Liu, K. Wang, J. Shi, Z. Qiao, and S. Shen, "Fm-fusion: Instance-aware semantic mapping boosted by vision-language foundation models," *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 2232–2239, 2024.

[15] N. Zimmerman, M. Sodano, E. Marks, J. Behley, and C. Stachniss, "Constructing metric-semantic maps using floor plan priors for long-term indoor localization," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 1366–1372.

[16] R. Martins, D. Bersan, M. F. Campos, and E. R. Nascimento, "Extending maps with semantic and contextual object information for robot navigation: a learning-based framework using visual and depth cues," *Journal of Intelligent & Robotic Systems*, vol. 99, no. 3, pp. 555–569, 2020.

[17] L. Xiao, J. Wang, X. Qiu, Z. Rong, and X. Zou, "Dynamic-slam: Semantic monocular visual localization and mapping based on deep learning in dynamic environment," *Robotics and Autonomous Systems*, vol. 117, pp. 1–16, 2019.

[18] N. Zimmerman, T. Guadagnino, X. Chen, J. Behley, and C. Stachniss, "Long-term localization using semantic cues in floor plan maps," *IEEE Robotics and Automation Letters*, vol. 8, no. 1, pp. 176–183, 2023.

[19] C. Huang, O. Mees, A. Zeng, and W. Burgard, "Visual language maps for robot navigation," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 10 608–10 615.

[20] E. Maiettini, V. Tikhanoff, and L. Natale, "Weakly-supervised object detection learning through human-robot interaction," in *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*, 2021, pp. 392–399.

[21] T. Ng, H. J. Kim, V. T. Lee, D. DeTone, T.-Y. Yang, T. Shen, E. Ilg, V. Balntas, K. Mikolajczyk, and C. Sweeney, "Ninjadesc: content-concealing visual descriptors via adversarial learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12 797–12 807.

[22] M. Nakanoya, S. S. Narasimhan, S. Bhat, A. Anemogiannis, A. Datta, S. Katti, S. Chinchali, and M. Pavone, "Co-design of communication and machine inference for cloud robotics," *Autonomous Robots*, vol. 47, no. 5, pp. 579–594, 2023.

[23] Z. He, T. Zhang, and R. B. Lee, "Model inversion attacks against collaborative inference," in *Proceedings of the 35th Annual Computer Security Applications Conference*, 2019, pp. 148–162.

[24] M. Luperto, M. Romeo, J. Monroy, J. Renoux, A. Vuono, F.-A. Moreno, J. Gonzalez-Jimenez, N. Basilico, and N. A. Borghese, "User feedback and remote supervision for assisted living with mobile robots: A field study in long-term autonomy," *Robotics and Autonomous Systems*, vol. 155, p. 104170, 2022.

[25] A. K. Taras, N. Suenderhauf, P. Corke, and D. G. Dansereau, "Inherently privacy-preserving vision for trustworthy autonomous systems: Needs and solutions," *Journal of Responsible Technology*, p. 100079, 2024.

[26] B. Liu, M. Ding, S. Shaham, W. Rahayu, F. Farokhi, and Z. Lin, "When machine learning meets privacy: A survey and outlook," *ACM Computing Surveys (CSUR)*, vol. 54, no. 2, pp. 1–36, 2021.

[27] X. Xu, D. Sun, J. Pan, Y. Zhang, H. Pfister, and M.-H. Yang, "Learning to super-resolve blurry face and text images," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 10 2017.

[28] S. Menon, A. Damian, S. Hu, N. Ravi, and C. Rudin, "Pulse: Self-supervised photo upsampling via latent space exploration of generative models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 2437–2445.

[29] R. Gilad-Bachrach, N. Dowlin, K. Laine, K. Lauter, M. Naehrig, and J. Wernsing, "Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2016, pp. 201–210.

[30] H. Chabanne, A. De Wargny, J. Milgram, C. Morel, and E. Prouff, "Privacy-preserving classification on deep neural network," *Cryptology ePrint Archive*, 2017.

[31] H. Hukkelås, R. Mester, and F. Lindseth, "Deepprivacy: A generative adversarial network for face anonymization," in *International symposium on visual computing*. Springer, 2019, pp. 565–578.

[32] X. Yu, K. Chinomi, T. Koshimizu, N. Nitta, Y. Ito, and N. Babaguchi, "Privacy protecting visual processing for secure video surveillance," in *2008 15th IEEE International Conference on Image Processing*. IEEE, 2008, pp. 1672–1675.

[33] M. U. Kim, H. Lee, H. J. Yang, and M. S. Ryoo, "Privacy-preserving robot vision with anonymized faces by extreme low resolution," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 462–467.

[34] J. Lin, Y. Li, and G. Yang, "Fpgan: Face de-identification method with generative adversarial networks for social robots," *Neural Networks*, vol. 133, pp. 132–147, 2021.

[35] Y. Wen, B. Liu, J. Cao, R. Xie, L. Song, and Z. Li, "Identitymask: deep motion flow guided reversible face video de-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 12, pp. 8353–8367, 2022.

[36] M. Ryoo, K. Kim, and H. Yang, "Extreme low resolution activity recognition with multi-siamese embedding learning," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 32, no. 1, 2018.

[37] M. Ryoo, B. Rothrock, C. Fleming, and H. J. Yang, "Privacy-preserving human activity recognition from extreme low resolution," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 31, no. 1, 2017.

[38] M. Gochoo, T.-H. Tan, F. Alnajjar, J.-W. Hsieh, and P.-Y. Chen, "Lownet: Privacy preserved ultra-low resolution posture image classification," in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. IEEE, 2020, pp. 663–667.

[39] A. S. Rajput, B. Raman, and J. Imran, "Privacy-preserving human action recognition as a remote cloud service using rgb-d sensors and deep cnn," *Expert Systems with Applications*, vol. 152, p. 113349, 2020.

[40] A. Li, J. Guo, H. Yang, F. D. Salim, and Y. Chen, "Deepobfuscator: Obfuscating intermediate representations with privacy-preserving adversarial learning on smartphones," in *Proceedings of the International Conference on Internet-of-Things Design and Implementation*, ser. IoTDI '21, 2021, p. 28–39.

[41] B. Azizian and I. V. Bajić, "Privacy-preserving autoencoder for collaborative object detection," *IEEE Transactions on Image Processing*, vol. 33, pp. 4937–4951, 2024.

[42] M. Li, X. Xu, H. Fan, P. Zhou, J. Liu, J.-W. Liu, J. Li, J. Keppo, M. Z. Shou, and S. Yan, "Stprivacy: Spatio-temporal privacy-preserving action recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 5106–5115.

[43] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *Proceedings of the IEEE*, vol. 111, no. 3, pp. 257–276, 2023.

[44] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *Advances in Neural Information Processing Systems*, vol. 28, no. 6, 2015.

[45] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," 2018.

[46] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," 2022.

[47] R. Balestriero and Y. Lecun, "How learning by reconstruction produces uninformative features for perception," in *Proceedings of the International Conference on Machine Learning (ICML)*, vol. 235, 2024, pp. 2566–2585.
[48] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.
[49] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, no. 3, pp. 211–218, 1936.
[50] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proceedings of the European Conference of Computer Vision (ECCV)*, 2014, pp. 740–755.
[51] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results," 2012.
[52] M. Antonazzi, M. Luperto, N. A. Borghese, and N. Basilico, "Development and adaptation of robotic vision in the real-world: the challenge of door detection," 2024.
[53] Z. Wang, E. Simoncelli, and A. Bovik, "Multiscale structural similarity for image quality assessment," in *Proceedings of the Thirty-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, vol. 2, 2003, pp. 1398–1402 Vol.2.
[54] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 586–595.
[55] J. Ichnowski, K. Chen, K. Dharmarajan, S. Adebola, M. Danielczuk, V. Mayoral-Vilches, N. Jha, H. Zhan, E. LLontop, D. Xu *et al.*, "Fogros2: An adaptive platform for cloud and fog robotics using ros 2," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 5493–5500.
[56] D. Vinod and P. SaiKrishna, "Development of an autonomous fog computing platform using control-theoretic approach for robot-vision applications," *Robotics and Autonomous Systems*, vol. 155, p. 104158, 2022.

Michele Antonazzi is a Ph.D. student working on domain adaptation and privacy preservation in Robotic Vision, with a particular focus on the paradigm of cloud robotics. He received his B.Sc. degree in Computer Science from the University of Padova in 2018 and his M.Sc. degree in Computer Science from the University of Milan in 2021.

Matteo Alberti earned his Master's degree in Cybersecurity at the University of Milan. His research interests include cybersecurity and artifical intelligence.

Alex Bassot received his M.Sc. degree in Mathematics from the University of Milan in 2024. He is currently a research fellowship in Robotics at the Department of Computer Science of the University of Milan, where he collaborates with the AIS-Lab. His research interests lie within the field of Multi-Agent Systems and Autonomous Robotics, with particular attention on Path Planning and Robotic Patrolling.

Matteo Luperto is an Assistant Professor at the Dipartimento di Informatica Giovanni Degli Antoni at the Università degli Studi di Milano (Italy). He received his Ph.D. in Information Technology from the Politecnico di Milano (Italy) in 2017. His main research interests lie within the field of semantic mapping for autonomous mobile robots in indoor environments, with a particular attention on the analysis of the structural properties of buildings, and long-term autonomy for social assistive mobile robots.

Nicola Basilico received his M.Sc. degree in Computer Science and Engineering in 2007 and his Ph.D. in Information Technology in 2011, both from Politecnico di Milano (Italy). He has held research positions as a postdoctoral scholar at the Robotics Laboratory of the University of California, Merced, and as a research assistant at the Swiss AI Lab IDSIA. From 2014 to 2019, he was an Assistant Professor, and since 2020 he has been an Associate Professor in the Department of Computer Science at the University of Milan (Italy). His research interests lie in Artificial Intelligence, with a particular focus on Multi-Agent Systems and Autonomous Robotics.