# PrivateEye: In-Sensor Privacy Preservation Through Optical Feature Separation

Adith Boloor *
Washington University in St. Louis
St. Louis, MO, USA
adith@wustl.edu

Weikai Lin
University of Rochester
Rochester, NY, USA
wlin33@ur.rochester.edu

Tianrui Ma
Washington University in St. Louis
St. Louis, MO, USA
tianrui.ma@wustl.edu

Yu Feng
Shanghai Jiao Tong University
Shanghai, China
y-feng@sjtu.edu.cn

Yuhao Zhu
University of Rochester
Rochester, New York, USA
yzhu@rochester.edu

Xuan Zhang
Northeastern University
Boston, MA, USA
xuan.zhang@northeastern.edu

## Abstract

*We address privacy issues in applications where images captured by an edge device (camera) are sent to the cloud for inference on utility tasks such as classification. Sending raw images to the cloud exposes them to data sniffing attacks and misuse by untrusted third-party service providers beyond the user's intended tasks. We propose an encoding scheme that not only evades direct visual inspection to the images or image reconstruction, but also prevents sensitive information from being ascertained. Unlike commonly used adversarial learning approaches, the proposed method is two-fold: first, it uses a diffractive optical neural network to spatially separate features corresponding to different tasks on the sensor plane in the optical domain. Then only the pixels corresponding to the utility task region are read. This encoding ensures that private features are never digitally stored on the edge device, thereby preventing privacy leakage. The proposed method successfully reduces the privacy retrieval in binary tasks with minimal accuracy loss ($\sim 2\%$) of the utility task, while reducing private task accuracy by $\sim 35\%$ and defending against reconstruction attacks with SSIM score of $0.43$.*

## 1. Introduction

We consider modern edge Computer Vision (CV) systems that are composed of a camera to capture images and send them to a cloud host to perform inference on deep learning models. However, these raw images often contain information beyond the user-authorized tasks (utility tasks), potentially exposing sensitive data that can be misused by untrusted third-party service providers (privacy tasks) [16, 35], leading to information leakage. Current privacy-focused commercial products use homomorphic encryption to secure the data [7], but it incurs substantial hardware overhead and slow computation speeds [9]. Additionally, storing captured images in the camera's digital memory makes them vulnerable to attacks [1, 20, 39].

As a solution, we propose PrivateEye, an optical system designed to encode images during acquisition (in the optical domain) to enhance visual privacy and prevent information leakage throughout the CV pipeline. PrivateEye features deliberate algorithm-hardware co-design. Algorithmically, the encoder is co-trained with downstream CV models to learn task-specific, privacy-preserving encodings. This ensures the encoded image contains only the features necessary for the intended utility task, while preventing a malicious third-party cloud host from recovering private information. At the hardware level, using an optical neural network [36, 45] as the encoder and an image sensor [6] as the sensing device, the proposed method prevents information leakage from the digital memory and improves efficiency by significantly reducing the amount of data sent to the cloud.

Fig. 1 provides an overview of our PrivateEye system, which comprises an edge device (client) and a cloud host (service provider). The overall goal is to learn an encoder at the edge that preserves features for utility tasks while inhibiting features related to private tasks for a given input data distribution. The encoding process involves two steps: feature separation and masking. During feature separation, we use a Diffractive Optical Neural Network (DONN) to spatially separate the features of the input image on the focal plane. For masking, we utilize the image sensor's native

*This project was funded by NSF grants NSF-2416375 and NSF-1942900.

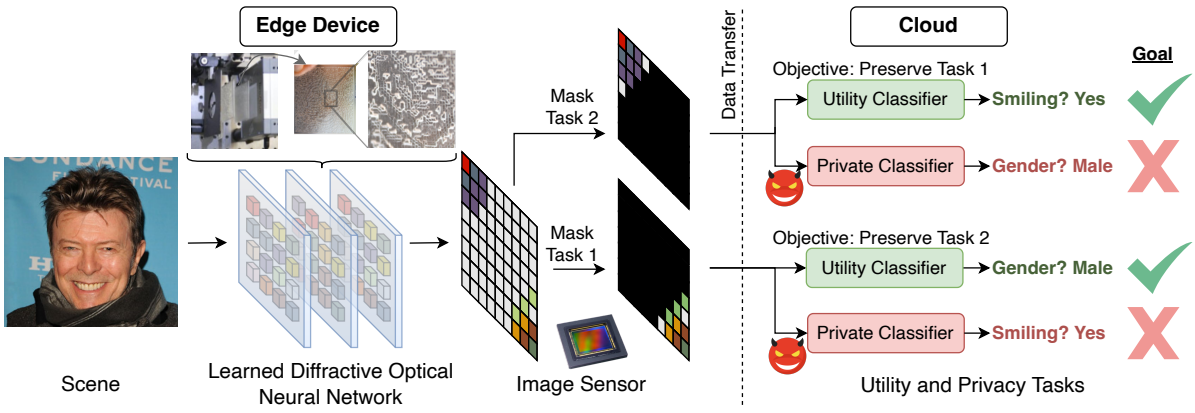


Figure 1. Conceptual overview of our pipeline: A 3D printable diffractive optical neural network is used to separate features belonging to different tasks (e.g., smiling and gender) and push them to the corners of the focal plane. Based on the user's choice of utility and private tasks, the image sensor masks out the corresponding pixels during image capture. The masked encodings are then sent to a third-party machine learning service provider for inference. A malicious attacker cannot recover private attributes from the encoded images.

pixel readout circuits to selectively read only a portion of the pixels while discarding the rest. This masking process retains the information necessary for the utility task while blocking information related to the privacy task. The key contributions of this paper are:

- Propose a privacy-preserving encoder at the optical-sensor level that trains a DONN to achieve spatial separation in the optical domain and masks privacy-dependent features during image readout.
- Design an anchor loss that separates task-specific features in the encoded image by pushing Class Activation Maps (CAMs) towards distinct spatial locations.
- Evaluate the privacy-utility trade-offs by testing the encoded images on various classifiers, ensuring that with aggressive masking (where 90% to 99% pixels are discarded), the private task accuracy is not recoverable while the utility task accuracy remains high.
- Demonstrate the DONN enables a near zero-energy privacy preservation encoding while providing great flexibility in the selection of privacy and utility tasks.

## 2. Background and Related Works

### 2.1. Imaging Pipeline for Vision Applications

Modern vision applications such as facial recognition, autonomous vehicle sign detection, and smart security systems depend on images captured through imaging processes. Fig. 1 illustrates how our imaging systems provide data for downstream vision tasks. In our setup, the imaging process involves optics and sensors: light from the scene undergoes optical processing before reaching the image sensor plane, where it is digitized for further processing.

Our work leverages optical computation and selective sensor readout to remove sensitive information early in the imaging pipeline. This prevents privacy leaks in the digital domain while also reducing computational overhead. To achieve this, we introduce a learnable DONN placed before the sensor, which is co-trained with the selection strategy and the CV models.

### 2.2. Privacy Preservation

**Digital Privacy Preservation with Neural Networks.** Several privacy-enhancing techniques have been proposed in neural networks including: obfuscation [37, 49], dimensionality reduction [28], and noise injection frameworks to enhance privacy when using third-party cloud services [15, 43]. Adversarial learning [29, 47, 48] and Generative Adversarial Models (GANs) have also been applied in this field. AdvPrivacy [29] suppresses private attributes while enhancing others, aiming to limit sensitive data recovery during training. DeepPrivacy [14, 25] masks sensitive facial features and generates new faces, while Cloak [26] removes non-task-specific features. However, these methods operate post-capture, making them vulnerable to pre-encoding attacks. In contrast, our method applies privacy-preserving measures before the image is captured, offering stronger security. Our approach is closely aligned with Adaptive Noise Injection (ANI) [15], which preserves privacy by selectively masking and adding noise to input data.

**Privacy Preservation in Optical Domain.** With the rapid advancement of optical computing [24, 36, 45, 50], privacy-preserving techniques are increasingly applied at the optical level to enhance security while reducing digital overhead. Bai et al. [3] demonstrated that learned diffractive lenses can project features from different data classes to distinct

imaging locations, enabling selective capture of target class images while erasing others. However, this method has only been tested on simple datasets (e.g., MNIST/Fashion-MNIST) likely due to the limitations of Optical Neural Networks (ONNs) in performing complex tasks, as they primarily perform linear transformations.

Building on previous work, lenses and phase masks have been designed for privacy in tasks like human action recognition, depth estimation, pose estimation, and facial de-identification [12, 13, 21, 38]. These often use adversarial training to balance utility and privacy [34]; however, these methods are task-specific, requiring retraining or lens reconfiguration [18] for different tasks or trade-offs. In contrast, our approach co-optimizes an ONN and sensor readout, enabling dynamic, post-fabrication control over task selection and utility-privacy trade-offs without reconfiguration.

### 2.3. Optical Computation

ONNs, which manipulate light waves instead of electrical signals, have attracted attention for their high-speed, low-energy computations. A typical ONN employs a *4-f* system with two convex lenses to perform optical convolutions through forward and inverse Fourier transforms [10]. By modulating the Fourier plane in both magnitude and phase, ONNs can effectively execute convolutions. Metasurfaces are commonly used to implement ONNs by controlling the physical parameters and orientations that define the Point Spread Function (PSF) of output light waves. This enables ONNs to perform downstream tasks like image classification [4, 30, 52] on datasets like MNIST.

DONNs are one of the most popular ONNs, which utilize diffraction to perform optical computations. Each diffraction layer embeds a phase modulator, where trainable phase modulation is applied to the light signal. The forward process of light propagation between layers is typically modeled using Fresnel, Rayleigh-Sommerfield, or Fraunhofer approximations iteratively [17]. DONN architectures support single-wavelength input, color channels management with beam splitters, and optical skip connections. However, unlike digital neural networks, DONNs lack non-linearities, which limits their ability to tackle complex tasks.

To make DONNs more accessible to deep learning researchers, several frameworks have been developed to simulate optical forward and backward models within deep learning environments. These include Mathworks BeamLab [42], Meta-Imager [51], LightPipes [41], and Lightridge [17]. Lightridge, implemented in PyTorch, simulates diffractive optics with differentiability across multiple optical layers, similar to convolutions in neural networks. In our work, we use Lightridge as the optics optimization framework.

## 3. Methodology

### 3.1. Use Scenario and Threat Model

We target a system where an image sensor interacts with a third-party service provider (cloud host). In this system, an image $x$ captured by the image sensor is sent to the host for prediction on certain tasks. During image capture, $x$ is encoded to $x'$ through the DONN and masking for the specific utility task. The encoded image $x'$ is then sent to the host. Since our method performs pre-capture visual privacy, $x$ is never stored beyond the optical layers.

We consider a scenario involving a attacker (eg. malicious service provider) who has gained access to the $x'$. $x'$ received by the attacker has the same dimensions as $x$ but with 90% to 99% of the values masked (filled with zeros). The attacker has access to the utility classifiers and can train their own classifiers to try to ascertain private attributes in the encoded images. We assume they have the same access to public datasets as we do and can use them to train a private classifier after querying our encoder.

### 3.2. Problem Setup

The core idea of our approach is to spatially separate task-specific features in the optical domain, enabling the use of task-specific masks to filter utility and private information. To illustrate our method, we consider a scenario with two tasks, `T1` and `T2` ($\texttt{T1}, \texttt{T2} \in \texttt{T}$), where `T` is the set of all possible tasks. Both `T1` and `T2` perform binary classifications with `T1` performing smile detection and `T2` detecting gender as seen in Fig. 1. PrivateEye supports scenarios with multiple tasks and various downstream CV models, which we analyze in Sec. 4,

Initially, we train the feature separation to ensure it effectively propagates features relevant to both tasks, enabling high classification accuracy for each. Once the encoder is trained, we apply a mask to retain features relevant to `T1` while disregarding those related to `T2`, based on user-defined settings that designate `T1` as the utility task and `T2` as the private task, or vice versa.

Our goal is to achieve high accuracy for `T1` and low accuracy for `T2`. To evaluate privacy preservation, the attacker trains a classifier with our learned, frozen encoder on public datasets. Our encoder is considered to be private if the attacker cannot substantially recover the accuracy for `T2`. Another aspect of our approach is its flexibility: we can select the private/utility task *after* training the encoder by modifying the mask pattern, which can be easily realized by programming the image sensor's readout sequence.

### 3.3. Training the Encoder

We begin by training binary classifiers for tasks `T1` and `T2` without any encoding. Subsequently, we train our encoder to spatially separate task-specific features. To achieve

this, we exploit Class Activation Maps (CAMs) [33, 53] to identify key regions on the image.

#### 3.3.1 Class Activation Maps and Anchoring

In our proposed solution, we spatially allocate features corresponding to `T1` to the top-left and `T2` to the bottom-right of the encoded space. This involves: (1) identifying task-specific features using CAMs and (2) guiding these features to specific locations through feature separation. CAMs are effective in identifying regions in an image that are most correlated to a classifier's output. We adopt the CAM notation directly from Grad-CAM [33], which uses the gradient information flowing into the last convolutional layer of a Convolutional Neural Network (CNN) to assign importance values corresponding to a particular decision.

To calculate the CAM $L^c_{Grad-CAM} \in \mathbb{R}^{u \times v}$ of height $u$ and width $v$ for a class $c$, the gradient of the score $y_c$ (pre-softmax) with the feature map activations $A^k \in \mathbb{R}^{k \times u \times v}$ of the convolutional layer is first computed, resulting in $\frac{\delta y^c}{\delta A^k}$. These gradients are global-average-pooled across their spatial dimensions to obtain neuron importance $\alpha^c_k$:

$$\alpha^c_k = \frac{1}{Z} \sum_i \sum_j \frac{\delta y^c}{\delta A^k_{ij}} \tag{1}$$

In Eq. (1), $\alpha^c_k$ captures the importance of feature map $k$ for class $c$. The weighted combination of the activation maps, followed by ReLU [27], is then performed to get CAM:

$$L^c_{CAM} = ReLU(\sum_k \alpha_k A^k) \tag{2}$$

ReLU is used to preserve only the positively correlated features. The produced CAM typically ranges from $4 \times 4$ to $16 \times 16$ for images of size 32 and 128, respectively, on ResNet-like models [11]. The CAM is usually upsampled to the image size and overlaid on the image to form a heatmap.

For our problem, an input image $x$ is processed through the encoder $f_e$ to obtain an encoded image $x'$. We have two classifiers for `T1` and `T2`, taking $x'$ as input and returning logits $y_1$, $y_2$ and activation maps $A_1$, $A_2$ for each task:

$$y_1, A_1 = f_{T1}(x'), \quad y_2, A_2 = f_{T2}(x') \tag{3}$$

We first calculate the CAMs using Eq. (1) and Eq. (2) and get CAMs for `T1`:

$$L_{CAM,T1} = |\sum_k \alpha_k A^k_{T1}| \tag{4}$$

Since we care about the entire binary task and not just a particular class, we take the absolute value of the CAM rather than ReLU. Similarly, we calculate $L_{CAM,T2}$. Note that for multi-class cases, we use the mean CAM across all classes.

Next, we define a weighted Euclidean distance between the calculated CAMs and specific "anchor" points $P \in \mathbb{R}^2$ located at the corners of the focal plane (e.g., $P = (0, 1)$ corresponds to top-right of the CAM). Now we maximize the separation between the two tasks, which we model using an anchor loss:

$$loss_{anchor}(L_{CAM}, P) = \sum_i^u \sum_j^v L_{CAM,ij} \sqrt{(i - P_0)^2 + (j - P_1)^2} \tag{5}$$

We normalize $L_{CAM}$ before using it in the loss function to prevent the optimizer from simply reducing the magnitude of the CAMs as a shortcut to minimize the overall loss. For example, we use this anchor loss to pull `T1` and `T2` features captured by pushing the CAMs to the top-left and bottom-right corners of the image respectively. To do so, we set $P_{T1} = (0, 0)$ for the top-left and $P_{T2} = (1, 1)$ to the bottom-right as indicated in Eq. (6):

$$\begin{aligned} \mathcal{L}_{anchor,T1} &= loss_{anchor}(L_{CAM,T1}, P_{T1}) \\ \mathcal{L}_{anchor,T2} &= loss_{anchor}(L_{CAM,T2}, P_{T2}) \\ \mathcal{L}_{anchor} &= \mathcal{L}_{anchor,T1} + \mathcal{L}_{anchor,T2} \end{aligned} \tag{6}$$

The general formulation for multiple $N$ tasks is $\mathcal{L}_{anchor} = \sum_i^N \mathcal{L}_{anchor,T_i}$ for various anchors $P$.

#### 3.3.2 Learning Regime

Since we have to optimize multiple networks (the encoder and $N$ classifiers), we first train the classifiers without any encoding so that they generate high quality CAMs. This also provides us the baseline accuracy. Then we freeze them and train the encoder using the objective function that combines the classifier loss and anchor loss:

$$loss = \sum_i^N \mathcal{H}(y_i, \tilde{y_i}) + \lambda \mathcal{L}_{anchor} \tag{7}$$

where $(y_i, \tilde{y_i})$ is the cross-entropy loss for task $i$ and $\lambda$ is a hyperparameter to control the strength of the anchor loss.

#### 3.3.3 Masking

Since the DONN is a physical component, once it is trained for a specific subset of tasks, it is fixed. However, we can program the mask onto an image sensor by selectively capturing only the unmasked pixels. Since we push features to the corners of the encoded space, the mask should reflect this. Therefore, we use a (rotated) upper diagonal 2D matrix mask, allowing values above the principal axis to pass through while blocking the rest. This has two advantages: 1) Trade-off Control: we can adjust the trade-off between the private and utility tasks by simply moving the principal

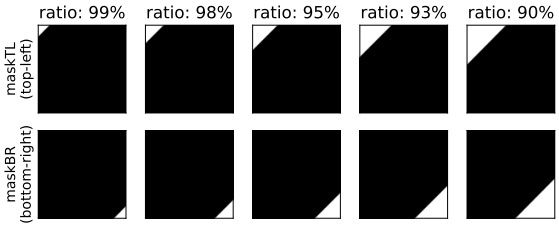


Figure 2. Different mask orientations and ratios control the privacy-utility trade-offs between tasks.

diagonal; 2) Energy Efficiency: by masking out a large portion of the encoded image, we capture significantly fewer pixels, directly improving the image sensor's energy efficiency. We generate and use different *corner* masks like $M_{top-left}$ and $M_{bottom-right}$ as shown in Figure 2 where masking ratios control the utility/privacy trade-off. Later, in Sec 4.3.4 we explore learned masking strategies.

### 3.4. Challenges with DONN

We test our hypothesis on a more traditional pixel-to-pixel model (e.g., UNet [31]) to verify that our objective function and training method produce separable CAMs. Table 1 compares the parameter sizes of various UNet and DONN architecture variants. The parameter size of UNet remains constant with input dimensions since UNet is a convolutional neural network. We control the size of UNet by adjusting the number of filters in each convolutional layer.

For the DONN, the table shows three variants with 10, 5, and 3 layers, respectively. The size of the diffractive layers increases with the image resolution, leading to different performances of the DONN architectures across various image scales. In Sec. 4, we experiment with UNet-tiny, and the DONN variants, excluding UNet-standard, as its parameter spaces exceed what the DONN models can replicate. Additionally, using too many DONN layers would be physically impractical. It is important to note that despite the DONN and UNet-tiny having similar parameter numbers, we do not expect the DONN to match the performance of UNet-tiny due to significantly different architectures, such as the lack of batch/layer normalization, multiple convolutional filters and non-linearities. Nonetheless, we demonstrate that we can achieve a trade-off between utility and private tasks, albeit with some performance degradation compared to UNet.

In practice, we observe that training the DONN directly with the task separation objective leads to poor convergence. To address this issue, we employ a student-teacher distillation approach where a UNet-tiny network serves as the teacher for a DONN network. To further alleviate the burden of replicating the entire UNet encoding function, we mask out part of the encoded features of both the student and the teacher, as shown in Eq. (8):

$$\begin{aligned} M &= M_{top-left} + M_{bottom-right} \\ loss &= MSE(x'_{student} \times M, x'_{teacher} \times M) \end{aligned} \tag{8}$$

This allows the DONN to focus only on learning the unmasked corner regions.

Table 1. Parameter size of various encoders. For UNet-tiny, we decrease the number of convolutional filters by a factor of 8.

| Encoder | #parameters (M) | | |
|---|---|---|---|
| | input 32×32×3 | input 128×128×3 | input 200×200×3 |
| UNet-standard | 31.04 | 31.04 | 31.04 |
| UNet-tiny | 0.49 | 0.49 | 0.49 |
| DONN-10 | 0.03 | 0.54 | 1.32 |
| DONN-5 | 0.02 | 0.29 | 0.72 |
| DONN-3 | 0.01 | 0.20 | 0.48 |

## 4. Experimental Evaluation

### 4.1. Datasets, Metrics, and Models

We primarily use the CelebA dataset [19] consisting of human faces and 40 labelled attributes of which we use a subset of. Similar to ANI [15], we focus on the smiling and gender tasks. We train models on two image resolutions to ensure the scalability of our model: $32 \times 32$ and $128 \times 128$. We use accuracy to compare the performance of `T1` and `T2`. To better gauge the performance of various encoders by their accuracies at different masking ratios, we introduce three metrics to quantify our results more precisely: 1) $\Delta_{util}$: utility accuracy loss due to encoding, characterized by $mean((acc_{base} - acc_{util}) * ratio_{mask})$; 2) $\Delta_{priv}$: attacker's ability to recover private information, characterized by $mean((acc_{priv} - acc_{rand}) * ratio_{mask})$; 3) $\Delta_{trade-off}$: quantifies the privacy-utility trade-off, given by $\Delta_{util} + \beta\Delta_{priv}$ where $\beta$ controls the importance given to privacy.

Each task uses a separate classifier, but they share a single encoder. For the classifiers, we use ResNet18 during training. During validation (i.e., when mimicking the role of the attacker), we experiment with various models including MLP-Mixer [40], ResNet [11], and ViT [5]. As described in Sec. 3.4, we use UNet-tiny as a digitally realized architecture and DONN, which is a diffractive neural network from the Lightridge framework for the encoders. We also explore and evaluate advanced architectures using like residual connections.

### 4.2. Training details

We first train each classifier for its respective task without any encoding to ensure a high baseline accuracy and the ability to generate high quality CAMs. Next, we select

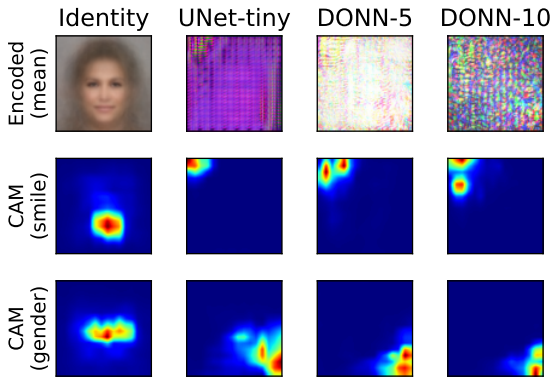


Figure 3. Visualization of CAMs from various encoders on $128 \times 128$ images in CelebA dataset. Rows from top to bottom are the mean of the encoded images, `CAMs-T1` (smiling), and `CAMs-T2` (gender), respectively. The proposed objective function clearly pushes the CAMs to the respective (top-left for `T1`, and bottom-right for `T2`) corners.

an encoder and train the entire pipeline using the learning regimes detailed in Sec. 3.3. We then freeze the encoder and mask the encoded regions using corner masks as descried in Sec 3.3.3. Finally, we train classifiers on the encoded images and report the accuracy on `T1` and `T2`. Our encoders are trained using the AdamW [23] optimizer, and the classifiers are trained using Stochastic Gradient Descent [32] with a learning rate of $1e^{-3}$ and momentum of 0.9. We use a Cosine Annealing [22] (without restarts) scheduler to decrease the learning rate over 100 epochs.

## 4.3. Results

We first assess the effect of our method in feature separation, then compare it against other encoding schemes. We then examine the robustness of our encoder to image reconstruction attacks. Finally, we conduct a series of ablation studies to analyze the impact of various characteristics and training methods. Additional results including more than two tasks, a face identification case study, and more ablation results can be found in our Supplementary material.

### 4.3.1 Verification of Feature Separation

We evaluate the setup in Sec. 3 by training `T1` and `T2` with an encoder that has learned to perform spatial separation and masking with increasing masking ratios. Fig. 3 shows that we can indeed train encoders to effectively separate the CAM regions for individual tasks with minimal performance degradation (without masks). Fig. 3(left-column) displays the default CAM regions for the two tasks on trained classifiers: `CAMs-T1` (smile) are centered around the mouth, while `CAMs-T2` (gender) are more spread out. Our encoded models successfully push the CAMs to the top-left and bottom-right regions, respectively. The top row visualizes the average encoded images for reference. We also evaluate different models in Table 2. Once an encoder is trained, we freeze it and train models for `T1` and `T2`, with utility and privacy tasks described by ↑ and ↓ respectively. The best trade-off is achieved with the original models (ResNet-18) used for training the encoder. So, we report our attack performance on ResNet-18 going forward.

Table 2. Utility ↑ and Privacy ↓ task performance using the top-left (maskTL) and bottom-right (maskBR) masking strategies.

| Model | Optimizer | MaskTL | | MaskBR | |
|---|---|---|---|---|---|
| | | T1↑ | T2↓ | T1↓ | T2↑ |
| MLP-Mixer | AdamW | 90.85 | 64.83 | 60.48 | 96.20 |
| MLP-Mixer | SGD | 90.80 | 56.13 | 58.23 | 96.15 |
| ResNet-18 | AdamW | 90.95 | 66.93 | 60.28 | 96.20 |
| ResNet-18 | SGD | 90.70 | 67.13 | 60.73 | 96.20 |
| ViT | AdamW | 90.60 | 56.13 | 58.23 | 96.10 |

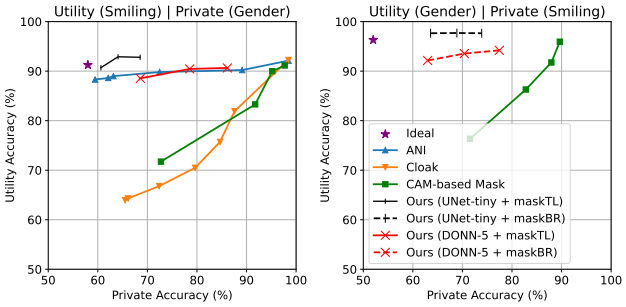


Figure 4. Comparison with other methods. (left) utility task is smiling and private task is gender; (right) tasks are flipped.

### 4.3.2 Comparison with other methods

**Privacy-Utility Trade-off.** In Fig. 4, we compare our method against other encoding-based models, such as ANI and Cloak [26], on the smiling (utility) vs. gender (private) task as presented in ANI. *Ideal* represents performance when the utility accuracy is that of a pretrained classifier without an encoder, and the private accuracy is equal to percentage of the largest class (i.e., random guessing) which is 52% for *smiling* task, and 58.1% for *gender*. Our method employs a DONN-5-res encoder, which is a 5-layer DONN with a residual connection, trained to separate `T1` and `T2` features. ANI controls their trade-off by adjusting the standard deviation of Gaussian noise added to the encoded images. We control it by changing the masking ratios of the top-left corner (maskTL) as shown in Fig. 2. For the set of task where the utility is smiling and privacy is gender (left), ANI achieves a good trade-off, maintaining utility accuracy close to the ideal (baseline) accuracy while the private accuracy decreases with increased noise magnitude.

Table 3. Comparison with optical baseline. T1, T2, T3, T4 are smiling, gender, lipstick, mouth open respectively. Bold and underlined text denotes best and second best accuracy in that column respectively. ↑ and ↓ are utility and private tasks respectively.

| | Scenario 1 | | Scenario 2 | | Scenario 3 | |
|---|---|---|---|---|---|---|
| **Method** | **T1↑** | **T2↓** | **T1↑** | **T3↓** | **T4↑** | **T2↓** |
| PPIA-GAP | 85.2 | 75.2 | 85.6 | 70.1 | 78.0 | 72.2 |
| PPIA-IS | **86.0** | 78.9 | **86.8** | 80.5 | 78.5 | 72.6 |
| PrivateEye | 85.4 | **62.7** | 82.4 | **58.2** | **79.1** | **62.6** |

In contrast, Cloak exhibits a poor trade-off, with a slope of close to 1, which indicates that irrespective of the parameter used to control the trade-off, it sacrifices too much utility accuracy to achieve low private accuracy. The UNet-tiny encoder performs better than DONN, in some cases beating the ideal utility accuracy. This is due to UNet's sophisticated architecture (as described in Sec 3.4) increasing the capability of the classifier.

The key part of our method is the feature separation followed by masking. Without this feature separation step, i.e., if we simply mask regions guided by the CAM heatmaps – the potential overlap between the utility and private task features could cause utility information to be lost when masked. We validate this by comparing our method against a purely CAM-based masking strategy (without a DONN feature separator). Fig. 4 shows that the CAM-based masking method has a poor trade-off compared to our method.

Finally, we showcase the flexibility of our method, where we can use the same DONN encoder and merely flip the mask to the bottom right (maskBR) and attain high performance on gender and low private accuracy on smiling as seen in Fig. 4 (right). ANI and Cloak are unable to do this without having to train a new encoder for this pair of task.

Furthermore, Table 3 demonstrates our encoders on various tasks against an optical encoder based PPIA [34] method. PPIA uses two different approaches to learn a single optical layer: GAP (generative adversarial privacy) and IS (Inverse Siamese). We observe that in most of these tasks, our approach significantly decreases the attainable private accuracy (denoted by ↓). In contrast to our other reported results, Table 3 uses a grayscale 64 × 64 size CelebA input to the optical encoder, and uses a MobileNetv2 classifier for all tasks to match the setting of PPIA.

#### 4.3.3 Reconstruction-Based Attacks

Until now, we have used private accuracy to gauge how well our method mitigates an attacker's ability to discern sensitive attributes. Now, we consider a different mode of attack: reconstruction-based attacks, wherein an attacker trains a network (e.g., UNet) to reconstruct the original images from the encoded images. To quantitatively measure the difference between the reconstructed and original images, we use the average Structural Similarity Index Measure (SSIM).


Figure 5. Reconstruction based attacks, along with average SSIM scores at various masking ratios.

To train the reconstruction network, we freeze the learned encoder, change various masking ratios, and train a full UNet model with a mean square error loss between the input and reconstructed images. Fig. 5 visualizes the reconstruction capability of an adversary based on different masking ratios. We observe that while the reconstruction recovers low frequency information such as color and overall structure, the attacker is unable to reconstruct key features, and even swaps genders in some reconstructions (e.g., the second row). The SSIM scores for each masking ratio also demonstrate that an attacker finds it extremely difficult to reconstruct the original images, with the difficulty increasing with the aggressiveness of the masking ratios.

#### 4.3.4 Ablation Study

**Encoder Architecture and Depth.** A DONN can be configured in various ways. Here, we explore the relationship between the number of diffractive layers – which characterizes physical realizability of the setup, and the utility-privacy trade-off. Fig. 6 visualizes the performance of various architectures: the left shows the privacy-utility curves for various architectures and the right shows the aggregated performance using the metrics proposed in Sec. 4.1. "-res" refers to an architecture with a residual connection, and the number refers to the number of layers in the DONN. Deeper layers tend to have better (lower errors) trade-off, and the performance trend is similar to that seen in LightRidge. Our findings reveal that in Fig. 6 (right), *donn-3* has the worst trade-off and use its $\Delta$ scores to normalize the rest of the results. Meanwhile, *donn-10-res* has the best utility-privacy trade-off for this task by having the lowest $\Delta_{trade-off}$. Without the residual connection, *donn-10* performs worse than *donn-5* (higher $\Delta_{priv}$), probably because deeper lay-

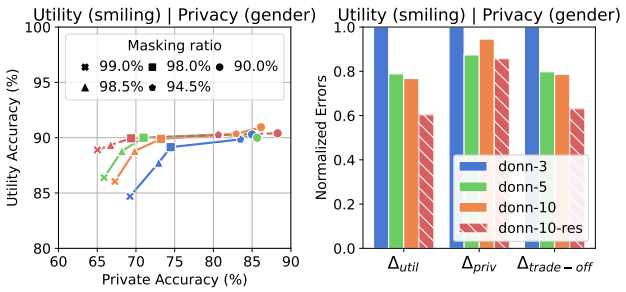


Figure 6. Comparative trade-off analysis contrasting DONNs with 3,5 and 10 layers, (and a residual version for donn-10).

Table 4. Learned masking on the T1: smiling, T2: gender task

| | T1 ↑ T2 ↓ | | T2 ↓ T1 ↑ | |
|---|---|---|---|---|
| **Masking Strategy** | **Acc T1** | **Acc T2** | **Acc T1** | **Acc T2** |
| Corner | 91.6 | 64.5 | 62.2 | 97.4 |
| Learned | 91.5 | **59.6** | **59.0** | 97.3 |

ers have more challenges converging to an optimum.

Residual connections on *donn-10-res* addresses this issue by allowing intermittent shortcuts from the input, which improves training. It is important to realize that while deeper layers perform better, they come with a physical realizability cost with having to fabricate more phase masks and use more lenses. Incorporating residual connections requires even more optical components like reflection mirrors [17]. Thus the cost of realizing the DONN architecture needs to be considered along with the desired trade-off.

**Learned Masking.** We initially mask the entire encoded image except for the corners to preserve `T1` or `T2` accuracy, but these static masks may be sub-optimal. To address this, we explore learned masks, hypothesizing that even within unmasked regions, pixels may leak sensitive information. We systematically learn the masks by freezing the encoder and adversarially training the mask to maximize utility and minimize privacy via the classifiers' cross-entropy losses.

We find that adversarially learning a full mask does not converge well. However, adding a corner mask prior, where the learned mask optimizes within a predefined region, leads to improved privacy protection. We learn the binary mask while enforcing an upper bound on the masking ratio. As shown in Table 4, with a masking ratio of 95%, the learned mask achieves a better trade-off than static corner masks. Although utility accuracy slightly decreases by ∼0.1%, private task accuracy is further reduced by ∼4% for the top-left corners and ∼3.2% for the bottom-right regions.

# 5. Discussion

## 5.1. Advantages of DONNs

**Lower Latency and Energy Consumption.** Due to the optical nature of DONNs, they exhibit minimal to no latency overhead compared to digital circuits. In contrast, incorporating a dedicated edge processor chip near the image sensor to handle conventional DNN encoding in the digital domain introduces significant overhead. Even an aggressive edge processor (systolic array with $32 \times 32$ MAC units and 7 $nm$ technology at 1 $GHz$) [8] consumes 5 $nJ/pixel$ and 13.6 $ms$ response time, as compared to 0.2 $pJ/pixel$ and 16.7 $ms$ allocated to image sensor itself ; less advanced edge processors would perform even worse. On the other hand, DONNs not only consume nearly zero energy and exhibit negligible latency, they also enable aggressive masking to reduce data volume, which further conserves energy and decreases latency on the image sensor hardware.

## 5.2. Limitations of DONNs

**Linearity of DONN.** Although the Fresnel approximation for light propagation appears non-linear due to Fourier transforms and trigonometric functions, the overall process remains linear with respect to the input light waves [2, 17, 46]. This means that our current DONN setup, which lacks non-linear activations, performs only linear transformations. We argue that: 1) For our current tasks, this linear setup provides a good privacy-utility trade-off; and 2) For more complex tasks, incorporating non-linear optical operations [44] could enhance performance. The images in the CelebA dataset are mostly well-aligned, with faces centered and exhibiting only minor variations. Our results demonstrate that the DONN can tolerate these slight variations and achieve a good utility-privacy trade-off. However, increased task complexity where images are not well-aligned poses a challenge for DONNs. We observe that while a UNet-tiny encoder can be trained using the PrivateEye framework provides a good privacy-utility balance and remains resilient to shifts of up to 8 pixels in the horizontal and/or vertical directions (for a $32 \times 32$ input), DONNs tend to leak more private information under similar conditions. We believe this issue remains to be fully addressed and leave it to future work.

Additional advantages and limitations of our method are discussed in the Supplementary material.

# 6. Conclusion

We propose PrivateEye, which employs a DONN to perform feature separation and a masking scheme to pass through only utility features to enhance privacy. By utilizing the optical nature of the DONN, the proposed method introduces nearly zero latency and energy overhead, while exhibiting superior privacy inhibition performance in various classification tasks.

## References


[1] Peshraw Ahmed Abdalla and Cihan Varol. Testing iot security: The case study of an ip camera. In *2020 8th International Symposium on Digital Forensics and Security (ISDFS)*, pages 1–5. IEEE, 2020. 1

[2] Claude Aime, E Aristidi, and Yves Rabbia. The fresnel diffraction: A story of light and darkness. *European Astronomical Society Publications Series*, 59:37–58, 2013. 8

[3] Bijie Bai, Yi Luo, Tianyi Gan, Jingtian Hu, Yuhang Li, Yifan Zhao, Deniz Mengu, Mona Jarrahi, and Aydogan Ozcan. To image, or not to image: class-specific diffractive cameras with all-optical erasure of undesired objects. *eLight*, 2(1):1–20, 2022. 2

[4] Julie Chang, Vincent Sitzmann, Xiong Dun, Wolfgang Heidrich, and Gordon Wetzstein. Hybrid optical-electronic convolutional neural networks with optimized diffractive optics for image classification. *Scientific reports*, 8(1):1–10, 2018. 3

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 5

[6] Abbas El Gamal and Helmy Eltoukhy. Cmos image sensors. *IEEE Circuits and Devices Magazine*, 21(3):6–20, 2005. 1

[7] Haokun Fang and Quan Qian. Privacy preserving machine learning with homomorphic encryption and federated learning. *Future Internet*, 13(4):94, 2021. 1

[8] Yu Feng, Tianrui Ma, Yuhao Zhu, and Xuan Zhang. Blisscam: Boosting eye tracking efficiency with learned in-sensor sparse sampling. *arXiv preprint arXiv:2404.15733*, 2024. 8

[9] Yanwei Gong, Xiaolin Chang, Jelena Mišić, Vojislav B Mišić, Jianhua Wang, and Haoran Zhu. Practical solutions in fully homomorphic encryption: a survey analyzing existing acceleration methods. *Cybersecurity*, 7(1):5, 2024. 1

[10] Joseph W Goodman. *Introduction to Fourier optics*. Roberts and Company publishers, 2005. 3

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4, 5

[12] Carlos Hinojosa, Miguel Marquez, Henry Arguello, Ehsan Adeli, Li Fei-Fei, and Juan Carlos Niebles. Privhar: Recognizing human actions from privacy-preserving lens. In *European Conference on Computer Vision*, pages 314–332. Springer, 2022. 3

[13] Carlos Hinojosa, Juan Carlos Niebles, and Henry Arguello. Learning privacy-preserving optics for human pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2573–2582, October 2021. 3

[14] Håkon Hukkelås, Rudolf Mester, and Frank Lindseth. Deepprivacy: A generative adversarial network for face anonymization. In *International symposium on visual computing*, pages 565–578. Springer, 2019. 2

[15] Sanjay Kariyappa, Ousmane Dia, and Moinuddin K Qureshi. Enabling inference privacy with adaptive noise injection. *arXiv preprint arXiv:2104.02261*, 2021. 2, 5

[16] Seong-Gyun Leem, Daniel Fulford, Jukka-Pekka Onnela, David Gard, and Carlos Busso. Not all features are equal: Selection of robust features for speech emotion recognition in noisy environments. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6447–6451. IEEE, 2022. 1

[17] Yingjie Li, Ruiyang Chen, Minhan Lou, Berardi Sensale-Rodriguez, Weilu Gao, and Cunxi Yu. Lightridge: An end-to-end agile design framework for diffractive optical neural networks. In *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 4*, ASPLOS '23, page 202–218, New York, NY, USA, 2024. Association for Computing Machinery. 3, 8

[18] Yingjie Li, Weilu Gao, and Cunxi Yu. Rubik's optical neural networks: Multi-task learning with physics-aware rotation architecture. In Edith Elkind, editor, *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, pages 7197–7206. International Joint Conferences on Artificial Intelligence Organization, 8 2023. Main Track. 3

[19] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Large-scale celebfaces attributes (celeba) dataset. *Retrieved August*, 15(2018):11, 2018. 5

[20] Yan Long, Qinhong Jiang, Chen Yan, Tobias Alam, Xiaoyu Ji, Wenyuan Xu, and Kevin Fu. Em eye: Characterizing electromagnetic side-channel eavesdropping on embedded cameras. *Proceedings of ACM NDSS*, 2024. 1

[21] Jhon Lopez, Carlos Hinojosa, Henry Arguello, and Bernard Ghanem. Privacy-preserving optics for enhancing protection in face de-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12120–12129, 2024. 3

[22] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 6

[23] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[24] Taiwei Lu, Shudong Wu, Xin Xu, and Francis TS Yu. Two-dimensional programmable optical neural network. *Applied optics*, 28(22):4908–4913, 1989. 2

[25] Maxim Maximov, Ismail Elezi, and Laura Leal-Taixé. Ciagan: Conditional identity anonymization generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5447–5456, 2020. 2

[26] Fatemehsadat Mireshghallah and et.al. Not all features are equal: Discovering essential features for preserving prediction privacy. In *Proceedings of the Web Conference 2021*, 2021. 2, 6

[27] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010. 4

[28] Seyed Ali Osia, Ali Shahin Shamsabadi, Sina Sajadmanesh, Ali Taheri, Kleomenis Katevas, Hamid R Rabiee, Nicholas D Lane, and Hamed Haddadi. A hybrid deep learning architecture for privacy-preserving mobile analytics. *IEEE Internet of Things Journal*, 7(5):4505–4518, 2020. 2

[29] Francesco Pittaluga, Sanjeev Koppal, and Ayan Chakrabarti. Learning privacy preserving encodings through adversarial training. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 791–799. IEEE, 2019. 2

[30] Geyang Qu, Guiyi Cai, Xinbo Sha, Qinmiao Chen, Jiaping Cheng, Yao Zhang, Jiecai Han, Qinghai Song, and Shumin Xiao. All-dielectric metasurface empowered optical-electronic hybrid neural networks. *Laser & Photonics Reviews*, 16(10):2100732, 2022. 3

[31] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015. 5

[32] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016. 6

[33] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 4

[34] Yamin Sepehri, Pedram Pad, Clément Kündig, Pascal Frossard, and L Andrea Dunbar. Privacy-preserving image acquisition for neural vision systems. *IEEE Transactions on Multimedia*, 25:6232–6244, 2022. 3, 7

[35] Animesh Srivastava, Puneet Jain, Soteris Demetriou, Landon P Cox, and Kyu-Han Kim. Camforensics: Understanding visual privacy leaks in the wild. In *Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems*, pages 1–13, 2017. 1

[36] Xiubao Sui, Qiuhao Wu, Jia Liu, Qian Chen, and Guohua Gu. A review of optical neural networks. *IEEE Access*, 8:70773–70783, 2020. 1, 2

[37] Qianru Sun, Liqian Ma, Seong Joon Oh, Luc Van Gool, Bernt Schiele, and Mario Fritz. Natural and effective obfuscation by head inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5050–5059, 2018. 2

[38] Zaid Tasneem, Giovanni Milione, Yi-Hsuan Tsai, Xiang Yu, Ashok Veeraraghavan, Manmohan Chandraker, and Francesco Pittaluga. Learning phase mask for privacy-preserving passive depth estimation. In *European Conference on Computer Vision*, pages 504–521. Springer, 2022. 3

[39] Ali Tekeoglu and Ali Saman Tosun. Investigating security and privacy of a cloud-based wireless ip camera: Netcam. In *2015 24th International Conference on Computer Communication and Networks (ICCCN)*, pages 1–6. IEEE, 2015. 1

[40] Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *Advances in neural information processing systems*, 34:24261–24272, 2021. 5

[41] Gleb Vdovin, Hedser van Brug, and Fred van Goor. Lightpipes: software for education in coherent optics. In *Fifth International Topical Meeting on Education and Training in Optics, Delft, The Netherlands*, pages 19–21, 1997. 3

[42] Madhu Veettikazhy, Anders Kragh Hansen, Dominik Marti, Stefan Mark Jensen, Anja Lykke Borre, Esben Ravn Andresen, Kishan Dholakia, and Peter Eskil Andersen. Bpm-matlab: an open-source optical propagation simulation tool in matlab. *Optics Express*, 29(8):11819–11832, 2021. 3

[43] Ji Wang, Jianguo Zhang, Weidong Bao, Xiaomin Zhu, Bokai Cao, and Philip S Yu. Not just privacy: Improving performance of private deep learning in mobile cloud. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2407–2416, 2018. 2

[44] Tianyu Wang and et al. Image sensing with multilayer nonlinear optical neural networks. *Nature Photonics*, 17(5), 2023. 8

[45] Tianyu Wang, Shi-Yuan Ma, Logan G Wright, Tatsuhiro Onodera, Brian C Richard, and Peter L McMahon. An optical neural network using less than 1 photon per multiplication. *Nature Communications*, 13(1):123, 2022. 1, 2

[46] John T Winthrop and CR Worthington. Convolution formulation of fresnel diffraction. *JOSA*, 56(5):588–591, 1966. 8

[47] Zhenyu Wu, Zhangyang Wang, Zhaowen Wang, and Hailin Jin. Towards privacy-preserving visual recognition via adversarial training: A pilot study. In *Proceedings of the European conference on computer vision (ECCV)*, pages 606–624, 2018. 2

[48] Taihong Xiao, Yi-Hsuan Tsai, Kihyuk Sohn, Manmohan Chandraker, and Ming-Hsuan Yang. Adversarial learning of privacy-preserving and task-oriented representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12434–12441, 2020. 2

[49] Kaiyu Yang, Jacqueline H Yau, Li Fei-Fei, Jia Deng, and Olga Russakovsky. A study of face obfuscation in imagenet. In *International Conference on Machine Learning*, pages 25313–25330. PMLR, 2022. 2

[50] Hui Zhang, Mile Gu, XD Jiang, Jayne Thompson, Hong Cai, Stefano Paesani, Raffaele Santagati, Anthony Laing, Y Zhang, Man-Hong Yung, et al. An optical neural chip for implementing complex-valued neural network. *Nature communications*, 12(1):457, 2021. 2

[51] Hanyu Zheng, Quan Liu, Ivan I Kravchenko, Xiaomeng Zhang, Yuankai Huo, and Jason G Valentine. Multichannel meta-imagers for accelerating machine vision. *Nature Nanotechnology*, pages 1–8, 2024. 3

[52] Hanyu Zheng, Quan Liu, You Zhou, Ivan I Kravchenko, Yuankai Huo, and Jason Valentine. Meta-optic accelerators for object classifiers. *Science Advances*, 8(30):eabo6410, 2022. 3

[53] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on*

*computer vision and pattern recognition*, pages 2921–2929, 2016. 4