ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

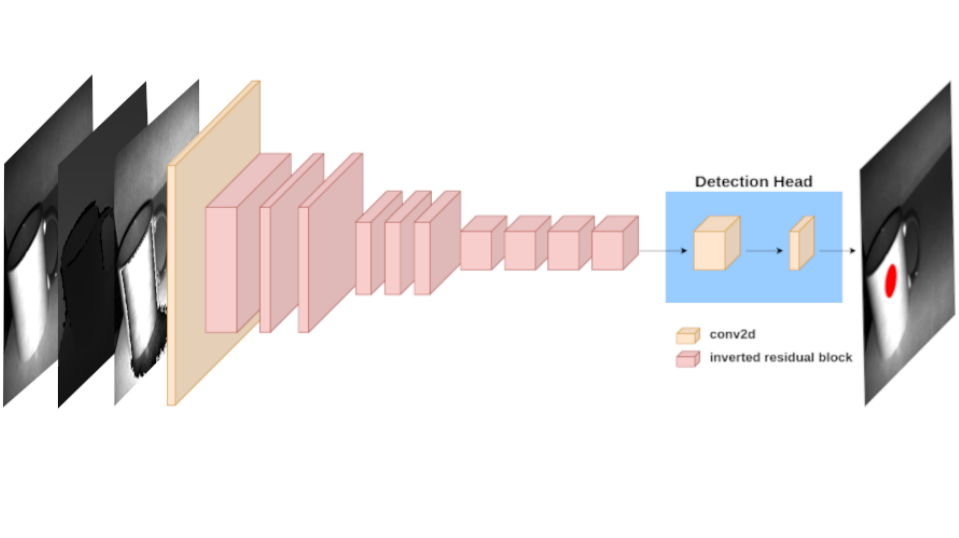
CENTER FOR PROJECT-BASED LEARNING

Department of Information Technology and Electrical Engineering

Autumn Semester 2023

Clear as Day: Low-Power Object Detection for Challenging Conditions

Semester Thesis



Jonas Bohn
bohnj@student.ethz.ch

February 2024

Supervisors: Hanna Müller, hanmuell@iis.ee.ethz.ch
Dr. Tommaso Polonelli, tommaso.polonelli@pbl.ee.ethz.ch
Professor: Prof. Dr. Luca Benini, lbenini@ethz.ch

Acknowledgements

I would like to express my sincere gratitude to my supervisors, Hanna Müller and Dr. Tommaso Polonelli, for their support, invaluable guidance, and continuous encouragement throughout this thesis.

I extend my heartfelt thanks to Elia, Fabio, Maurice, and Michel, whose active participation and contribution were pivotal in the creation of the dataset essential for my work. Their collaboration significantly enriched the depth and quality of my research, and I am truly grateful for their involvement.

Abstract

Understanding the surrounding environment is crucial for the autonomy and fluid interaction of robotic systems. Significant progress has been made in object detection through large-scale convolutional neural networks. However, these methods are often impractical for devices with limited computing resources. To mitigate this, research has addressed this problem by developing more efficient, scaled-down versions of networks like YOLO [1], enabling them to run in real time on resource-constrained devices. Despite these advances, most efforts concentrate on visual data from traditional cameras, which struggle under poor lighting or occlusions. Some systems have begun integrating multiple data sources to leverage their combined strengths and address these challenges, especially where computational resources are not a primary concern.

This work explores sensor fusion for object detection in challenging light conditions using the flexx2 3D camera. The focus is on integrating infrared and depth data to enhance object detection performance on resource-constrained and low-power devices, particularly in robotics and autonomous systems where efficiency and accuracy in object detection are crucial under varied environmental conditions. As part of this work, a novel dataset for object detection combining infrared and depth data is introduced, employing the Faster Objects, More Objects (FOMO) model for sensor fusion. The thesis showcases the feasibility of using sensor fusion and FOMO for fast and low-power object detection on constrained devices.

Leveraging sensor fusion with the FOMO model, this work highlights the potential for enhanced object detection in poor lighting via infrared and depth data integration, while minimally impacting the computational efficiency and power consumption on the GAP9 processor. The streamlined network, with only 18.2k parameters, achieves rapid inference times of 5.4 ms and consumes just 60 mW per frame on GAP9. This performance outpaces a simplified version of YOLO [2], delivering threefold faster predictions, 22 times fewer parameters, and 34 mW lower power consumption per frame, emphasizing sensor fusion's potential in low-power scenarios.

Declaration of Originality

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor. For a detailed version of the declaration of originality, please refer to Appendix B

Jonas Bohn,
Zurich, February 2024

Contents

List of Acronyms **ix**

1. Introduction **1**

2. Related Work **3**
2.1. Object Detection 3
2.1.1. Constrained Devices 3
2.2. Sensor Fusion for Object Detection 4

3. Methods **7**
3.1. Data Collection 7
3.1.1. Flexx2 Camera 7
3.1.2. Flexx2 Dataset 9
3.2. Object Detection 10
3.2.1. Data Preprocessing 11
3.2.2. Sensor Fusion 11
3.2.3. Data Augmentation 13
3.2.4. Network Architecture 14
3.3. Model Deployment 15

4. Results **17**
4.1. Training Setup 17
4.2. Fusion Experiments 18
4.3. Model Performance on GAP9 19

5. Discussion **23**
5.1. Impact of Depth in Detection 23
5.2. Impact of Depth on GAP9 Performance 24

6. Conclusion and Future Work **26**

A. Task Description **28**

B. Declaration of Originality **35**

List of Figures

2.1. Different sensor fusion approaches for RGB images and depth data 5

3.1. Outputs from flexx2 camera taken in the same setting 8
3.2. Infrared and grayscale depth map pairs of the recorded dataset 10
3.3. Distribution of objects in the dataset 11
3.4. Comparison of images before and after center cropping 12
3.5. Results of sensor fusion strategies with where the input channels are displayed using RGB color mapping 13
3.6. Different data augmentation techniques used, for better visibility only the IR channel was used except in the demonstration in 3.6f, where RGB channels are used to showcase the dropout of the R and B channels 15
3.7. FOMO network architecture 16
3.8. Example outputs for FOMO network 16

4.1. Qualitative results of the different fusion strategies 19
4.2. Power measurements on GAP9 over time 20
4.3. Power measurements for inference of the tested fusion approaches 21

5.1. Examples of positive effect of fusing depth data 24

List of Tables

2.1. A comparison of our work against state of the art lightweight object detection models 6

3.1. Camera modes and specifications of the flexx2 camera [3] 9

4.1. Comparison of different fusion performances 18
4.2. Comparison of Fusion Performances on GAP9 22

5.1. Comparison of fusion performances on GAP9 25

List of Acronyms

FOMO . . . . . Faster Objects, More Objects

FOV . . . . . Field of view

IIS . . . . . Integrated Systems Laboratory

IoT . . . . . Internet of Things

IR . . . . . Infrared

LiDAR . . . . . Light Detection and Ranging

PBL . . . . . Center for Project-Based Learning

ToF . . . . . Time-of-Flight

Chapter 1

Introduction

In the field of robotics, the quest for autonomy and smooth interaction with dynamic environments has become a focal point, driven by the current trend in deep learning methods. Learning-based techniques are becoming increasingly popular in the field of visual perception for example on robots or smart devices, which helps machines understand and respond effectively to their environment [4]. A core component of perceiving the environment lies in detecting objects present in the field of view of the robotic system. Whether navigating around obstacles or targeting specific objects, object detection is a fundamental component of seamless robotic intelligence and autonomy [4]. Computer vision research, initially driven by the development of deep convolutional networks [5], is constantly working on new network architectures to further improve the peak accuracies on benchmark datasets such as ImageNet [6], PASCALVOC [7] or COCO [8]. However, these networks are computationally and power-wise very expensive which makes them infeasible for use in real-time applications on mobile robots, where computation and power consumption are constrained [9]. The requirement for an object detection network in robotics is that it must perform quickly and reliably under challenging conditions, such as poor illumination, occlusion, or highly dynamic scene changes.

Most work on object detection is based on monocular image data captured in RGB color channels. However, only relying on camera data can limit the effectiveness of object detection, especially in challenging conditions such as low light or darkness. In such scenarios, where visual cues are scarce or nonexistent, the reliance on camera data alone may result in inadequate or inaccurate detection performance.

In recent years, there has been progress in fusing data streams from different sensors to improve performance and robustness in what is known as sensor fusion. In particular, laser-based sensors such as Light Detection and Ranging (LiDAR) or laser-based Time-of-Flight (ToF) sensors are often combined with a monocular camera to complement each other's strengths and overcome their limitations. Laser-based sensors, such as ToF, can

accurately measure depth information, which helps to segment objects within the scene and better understand the 3D environment around the mobile robot. Combining it with the camera data, the fused system is less affected by environmental factors such as low light or fog [10] and can therefore provide reliable data in those situations.

This thesis aims at sensor fusion for object detection in difficult light conditions using the flexx2 3D camera developed by pmdtechnologies. The flexx2 contains a laser-based ToF sensor that can output visual information in the form of an active infrared (IR) image and a 3D point cloud that can easily be converted into a 2D depth map. This particular ToF sensor allows depth and visual data to be fused for detection in the same resolution, which has been a bottleneck in previous work at PBL [11]. To further process the data to detect objects, a lightweight architecture called Faster Objects, More Objects [12] (FOMO), is used. FOMO is specifically designed to perform real-time object recognition on constrained devices. While FOMO is well suited for detection in RGB images, its architecture has not yet been applied to sensor fusion with infrared and depth data.

This work demonstrates the feasibility and benefits of using FOMO to perform sensor fusion with infrared and depth data captured by the flexx2 camera targeting resource-constrained and low-power devices with a network size of 18.2k parameters. To demonstrate this, a new object detection dataset is created containing active infrared and depth data with bounding box labels for three classes. It is also shown that the resulting fusion network can be quantized and ported to a constrained device achieving processing times of 5.4 ms per frame with an average power consumption of 60 mW.

The thesis first delves into existing sensor fusion techniques and object detection models. It progresses to detail the methodologies adopted for data collection, preprocessing, and fusion approaches for an object detection network. Experimentation discusses the setup and results of applying these approaches. The final chapters synthesize the findings, evaluate the success of the proposed model, and suggest directions for future research based on this thesis.

Chapter 2

Related Work

This section delves into the foundational and contemporary works related to object detection, highlighting the unique challenges and solutions when deploying these technologies on constrained devices. Additionally, the role of sensor fusion in enhancing object detection capabilities is explored, focusing on strategies for fusing visual and depth data.

2.1. Object Detection

The field of object detection has witnessed significant advancements in recent years [13], fueled by the success of deep convolutional networks in [5]. Techniques such as Region-based CNNs (R-CNNs) [14, 15, 16, 17], Single Shot MultiBox Detectors (SSDs) [18], and You Only Look Once (YOLO) [19] have shaped state-of-the-art performance in object detection. These networks, however, are typically designed to run on powerful GPUs and struggle to be applied on mobile robots or Internet of Things (IoT) devices [20, 9, 21]. The deployment of object detection on constrained devices, such as microprocessors, introduces a new set of challenges. The constraints inherent in these devices, including limited computational power, memory, and energy resources, pose unique challenges that necessitate novel approaches and tailored solutions.

2.1.1. Constrained Devices

A common strategy employed in object detection on edge devices involves optimizing well-established deep neural networks to meet the constraints of limited computational resources. Researchers have explored various techniques to achieve this optimization, focusing on reducing the size and computational requirements of the models [21, 22].

Quantization is one such technique that has gained prominence in the context of edge device deployment [23, 24, 25]. This method involves reducing the precision of the model's weights, effectively storing them in lower bit precision while striving to maintain acceptable accuracy levels. Li et al. [26] demonstrated that their approach to quantizing an object detection network from 32-bit to 4-bit achieves detection performances very closely to the 32-bit model, saving 8x of the memory.

In addition to quantization, pruning is another technique used for model compression in object detection applications [27, 28]. Pruning aims to eliminate unnecessary components from a pre-trained neural network, including weights, layers, connections, and neurons [29]. By removing redundant elements, the model becomes more compact and computationally efficient.

Beyond quantization and pruning, researchers have explored techniques such as knowledge distillation [30, 31], parameter reduction, and hybrid approaches that combine multiple compression techniques to achieve even greater model efficiency [9, 23]. Multiple works using hybrid approaches focus on combining or modifying the most influential object detection networks such as variants of MobileNet [32, 33], YOLO [19, 1, 34], EfficientNet [35] or SSDs [18] to construct new lightweight networks [36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 2].

In 2022, Edge Impulse, a commercial company specializing in deploying machine learning models on resource-constrained devices, unveiled an innovative detection model known as FOMO, as highlighted in their announcement [12]. This compact model demonstrated promising results by leveraging a modified MobileNetV2 [39] architecture for feature extraction. The FOMO model diverges by truncating the network after the initial layers, resulting in a coarse feature map. Instead of predicting precise bounding boxes from the feature map, FOMO focuses on estimating the centroids of detected objects. This approach empowers the model to achieve real-time detection of multiple objects, consuming up to 30 times less processing power and memory compared to MobileNet SSD or YOLOv5 [12].

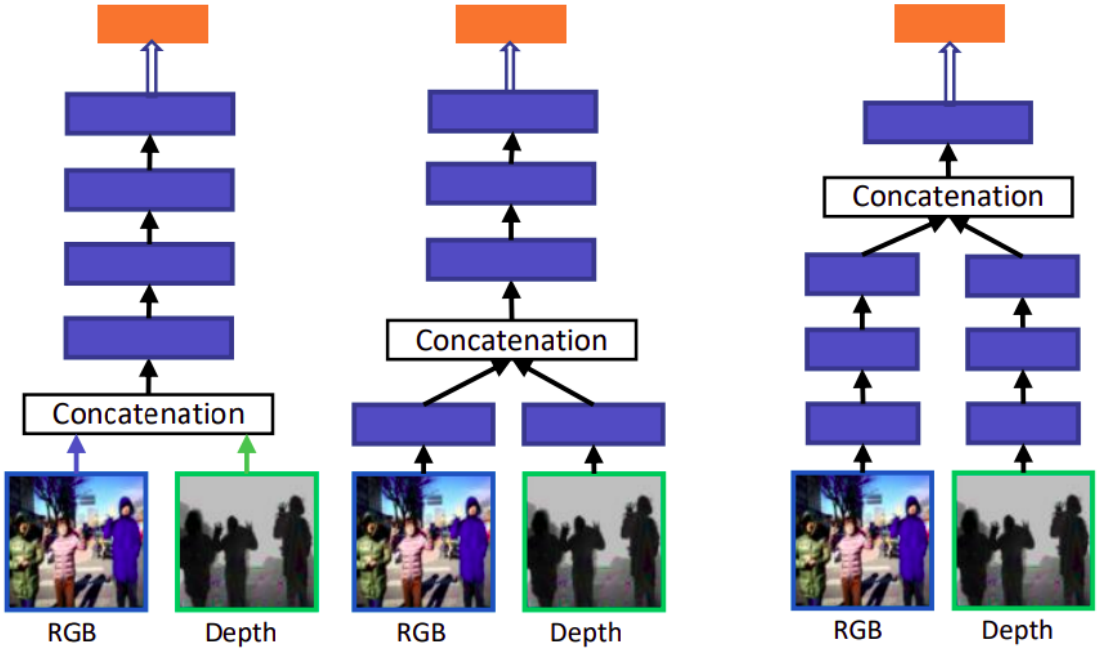
2.2. Sensor Fusion for Object Detection

Traditional object detection methods primarily rely on visual data, which encounter difficulties in various real-world scenarios, including occlusions, varying light conditions, and diverse environmental settings including fog, heavy rainfall, or snow. Sensor fusion seeks to harness synergies among different sensors, such as cameras, depth sensors, and LiDAR, to create a more nuanced and comprehensive representation of the surroundings. The integration of information from multiple sensors has proven to be instrumental in overcoming the limitations inherent in individual sensing modalities [47, 48, 49, 50, 10, 51, 52]. Current research on fusing visual and depth data for object detection can be

categorized into two large groups, distinguished by the specific point within the network where the fusion of data occurs.

Farahnakian et al. [47] used an early or feature-level fusion approach by concatenating RGB and depth data into a four-channel input, as seen in Figure 2.1a to a Faster R-CNN [17] based detection network, fusing directly the raw input data. Their input fusion approach demonstrated improved object detection performance and robustness compared to the unimodal approach containing only RGB or depth data on the KITTI dataset [53]. [54] states that this early input fusion requires the different modalities to be semantically similar to achieve good performances without computationally heavy preprocessing. Furthermore, Xu et al. [55] went beyond early input fusion, to apply pose estimation with early fusing data from an RGB camera and a LiDAR sensor after feeding each data stream into a small independent feature extraction network, demonstrated in Figure 2.1a.

In [49], the experimental findings indicate that the fusion of RGB and depth data in the later stages of their network leads to the best performance, called late or decision-level fusion as seen in Figure 2.1b. The authors point out, that there is no distinct optimal fusion level. They state that the fusion level should serve as an additional hyperparameter, which requires tuning for each unique case. Late fusion approaches perform classification or detection independently for each sensory input and then fuse the outputs at the decision level to achieve a final classification or detection.



(a) Early fusion [56]

(b) Late fusion [56]

Figure 2.1.: Different sensor fusion approaches for RGB images and depth data

Previous work [11, 57] explored the impact of using synthetic data in pretraining a sensor fusion network when data is scarce. Brander et. al [11] created a simulated dataset consisting of a 170x170 grayscale image and 8x8 ToF depth data, which showed to be an effective and cheap way of generating data for training a classification network. The work [11] showed, that pretraining the classification network on synthetic data and fine-tuning on real-world data resulted in an increased accuracy compared to no pretraining. Furthermore, six different approaches to fusing the data are compared, where the best performance is achieved in a later stage fusion, which happens to be because of the difference in sensory input size [54].

While the existing studies have delved into sensor fusion for improved perception, the integration of active IR and depth information on constrained devices remains an unexplored frontier. This thesis aims to address this gap by introducing a novel approach that leverages the synergies between active IR and depth data, optimizing object detection performance on devices with limited computational resources in difficult light conditions.

Table 2.1 summarizes the most important work in lightweight object detection for this thesis compared to our final model. It can be observed that most work either focuses on lightweight object detection or sensor fusion using very computationally expensive state-of-the-art object detection models. This work tries to combine both research areas while to the best of our knowledge, no scientific paper has yet investigated or proposed a similar fusion strategy in the context of object detection on resource-constrained devices.

Reference	**Fusion**	**Detection**	**InputSize (Px)**	**Parameters**
YOLOv8n [46]	×	✓	640	3.2M
YOLOX-Nano [45]	×	✓	416	0.9M
YOLOv5p [37]	×	✓	256×192	0.62M
Tiny YOLO-Lite [44]	×	✓	416	0.6M
TinyissimoYOLO [36]	×	✓	88	0.42M
Lamberti et al. [58]	×	✓	320x240	4.67M
Bijelic et al. [10]	7 sensors	✓	1920x1024	-
Brander et al.[11]	Gray, Depth	×	170, 8	0.126M
Ours	IR, Depth	✓	168, 168	0.018M

Table 2.1.: A comparison of our work against state of the art lightweight object detection models

Chapter 3

Methods

This section offers a detailed overview of the used methodology, encompassing the techniques employed for data collection, the deep learning procedures, and the strategies adopted for the effective deployment of the trained model onto constrained platforms.

3.1. Data Collection

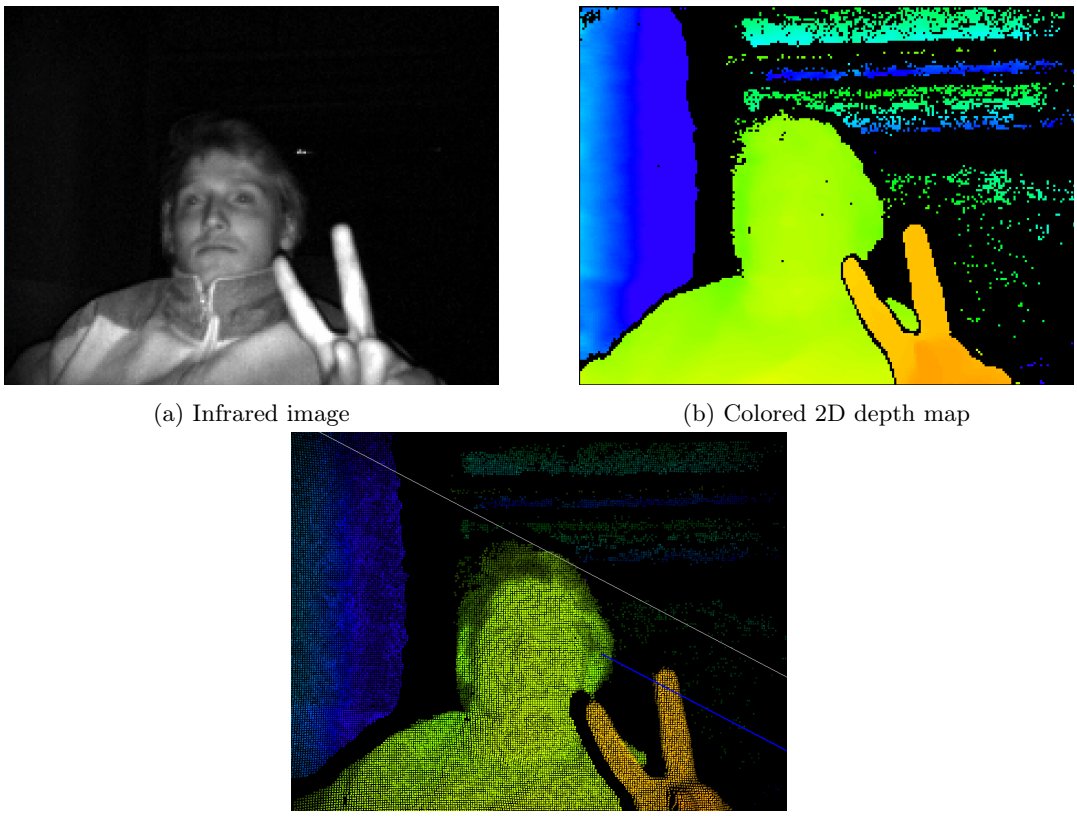
The data used for training the object detection dataset was captured using the flexx2 3D camera of pmdtechnologies, where Section 3.1.1 discusses the technology and specifications of the camera. Section 3.1.2 addresses the creation of the new dataset, which serves as a foundational resource for subsequent stages in this research, particularly in the training and evaluation of the object detection model.

3.1.1. Flexx2 Camera

The flexx2 camera is a laser-based ToF camera with a USB interface for power and data transfer. The camera weighs 13 grams and has an average power consumption of 300 mW [3]. At the core of the flexx2 camera is the IRS2381C REAL3™ Time-of-Flight Image Sensor from Infineon, a critical component responsible for capturing depth information. The sensor operates by emitting infrared light pulses and measuring the time it takes for these pulses to travel to the subject and back, which enables to calculation of the distance. By calculating the distance to each point in the scene, the sensor constructs a detailed representation of the environment in the form of a point cloud, shown in Figure 3.1c, which can additionally be represented as a 2D depth map as Figure 3.1b illustrates. In addition to depth information, the flexx2 produces active IR images in the same resolution and Field of view (FOV) as the point cloud or depth map depicted

in Figure 3.1a. Active IR illumination enables the camera to operate in various lighting conditions, making it versatile for both indoor and outdoor applications [3].

Unlike earlier studies at PBL [11, 37], where the resolution and FOV of the ToF depth map and the visual representation did not align seamlessly, the flexx2 camera addresses this limitation, presenting a significant advantage. The flexx2 camera ensures an overlapping resolution of 224 x 172 pixels for both the ToF depth map and the visual representation, enhancing the coherence and accuracy of the scene representation.



(a) Infrared image

(b) Colored 2D depth map

(c) 3D point cloud

Figure 3.1.: Outputs from flexx2 camera taken in the same setting

The flexx2 camera operates in nine different working modes as shown in Table 3.1, where the user can dynamically adjust both the distance and framerate parameters, tailoring the ToF camera's operation to suit the specific requirements of their application. According to the manufacturer, the "5" modes use about 30% less processing power than the "9" modes and are better suited for lower distances until a maximum of three meters and in cases where computation is a concern. While mode "5" focuses on fast processing, mode "9" focuses on a better depth map quality and longer ranges [3].

ModeName	Exposure Time (µS)	FPS	Approx. Max Range (indoor)
Mode "5" for faster processing, lower distances			
MODE_5_15FPS	1040	15	3 meters
MODE_5_30FPS	500	30	2 meters
MODE_5_45FPS	310	45	2.5 meters
MODE_5_60FPS	220	60	1 meter
Mode "9" for longer distance, better depthmap			
MODE_9_5FPS	1500	5	4 meters
MODE_9_10FPS	760	10	3.5 meters
MODE_9_15FPS	500	15	3 meters
MODE_9_20FPS	390	20	2.5 meters
MODE_9_30FPS	220	30	2 meters

Table 3.1.: Camera modes and specifications of the flexx2 camera [3]

The flexx2 camera comes with a software package called Royale, an easy camera framework for ToF cameras which enables fast integration and testing on various platforms. While the framework is designed in C++, a wrapper for Python does exist, which was used to create the dataset in Section 3.1.2.

3.1.2. Flexx2 Dataset

For the creation of a new dataset, computational time and fast FPS modes are deemed unnecessary; the priority lies in acquiring clean depth maps with minimal invalid pixels, hence the selection of mode "9" for superior depth quality. It's important to note that invalidating pixels or introducing noise can be easily managed post-creation. The MODE_9_10FPS is used in particular to capture the IR and depth data for training the object detection model from Section 3.2.4, where only every tenth frame is used in the dataset to account for more diverse scenes.

The collected dataset features three different object classes, which are all part of the ImageNet dataset [6]. Persons, cups, and apples are recorded in multiple different indoor scenarios, where some example images can be seen in Figure 3.2. The images show grayscale depth maps over RGB depth maps because they contain the same depth information but require less memory, offering an efficient alternative without compromising data quality. To account for a more robust and versatile object detector, different sized and shaped objects are used for data collection.

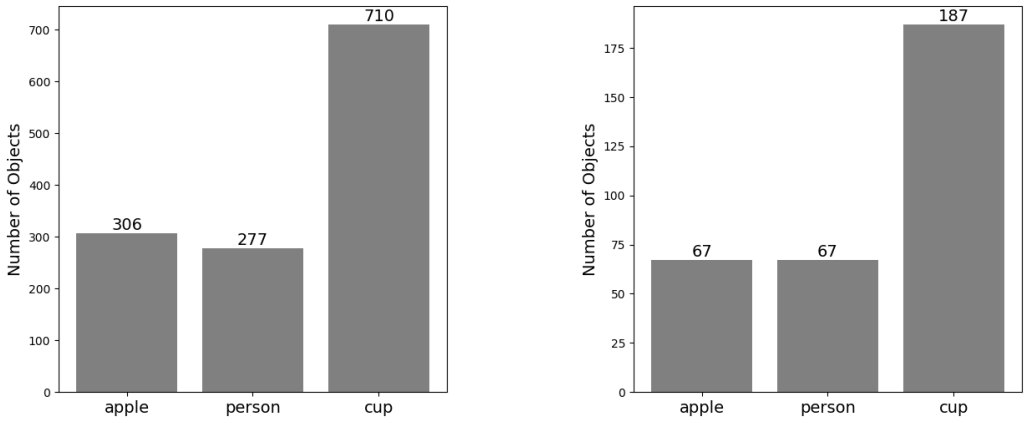
The total dataset consists of 1421 IR and depth map pairs in a resolution of 224 x 172 pixels and is labeled using the annotation tool by Edge Impulse [59]. The resulting dataset is divided into two subsets, allocating 80% for training purposes and reserving the remaining 20% for testing. Figure 3.3 showcases the distribution of class labels

amongst training and testing datasets, where the imbalance of cups comes due to practical limitations and resource constraints during the data labeling phase. It is important to highlight that this imbalance was unintentional. Despite these challenges, the available dataset serves as a valuable foundation for further exploration and analysis.

Figure 3.2.: Infrared and grayscale depth map pairs of the recorded dataset

3.2. Object Detection

Prior to utilizing the created dataset for training, the data has to be preprocessed to ensure the optimal format, which is covered in Section 3.2.1. Following preprocessing, various data fusion methods are explored, which are introduced in Section 3.2.2, before training the object detection model. The architecture of the used network is thoroughly described in Section 3.2.4. However, before delving into the architecture specifics, the data augmentation strategies employed are discussed in Section 3.2.3, which are instrumental in enhancing the model's performance under challenging conditions.



(a) Distribution of objects in training data

(b) Distribution of objects in testing data

Figure 3.3.: Distribution of objects in the dataset

3.2.1. Data Preprocessing

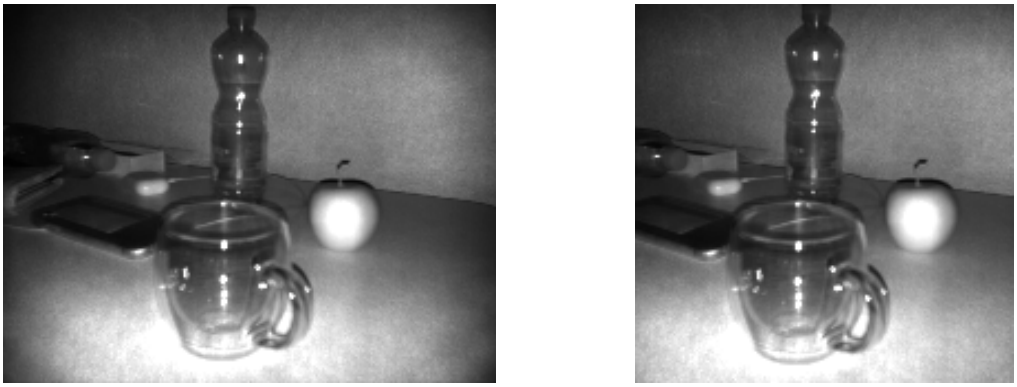
In preparation for training an object detection model, a critical aspect of the data preprocessing pipeline involves transforming the input images to meet the expectations of the model and facilitate computation in training the network.

The underlying object detection model, which will be introduced in more detail in Section 3.2.4, necessitates square input dimensions to work properly. To achieve the required input dimensions, a center crop operation is applied to the original images which results in a square image set with dimensions of 168 x 168 pixels. Figure 3.4 demonstrates the effect of center cropping on the original dataset, where data is lost on the edges of the image.

3.2.2. Sensor Fusion

In the context of the used object detection framework, sensor fusion plays a pivotal role in leveraging multiple modalities for enhanced perception. The integration of IR and depth data are explored through early fusion techniques, motivated by insights from the relevant literature in Section 2.2.

To evaluate the effectiveness of the proposed fusion strategies, they are compared against a baseline model that uses only a single channel of IR data. This baseline offers insights into the relative performance gains achieved through the integration of depth information and the exploration of different early fusion strategies. Specifically, three distinct early



(a) Before center crop (b) After center crop

Figure 3.4.: Comparison of images before and after center cropping

fusion strategies are considered for object detection, involving the incorporation of the preprocessed IR and depth data.

(IR, Depth, Depth)

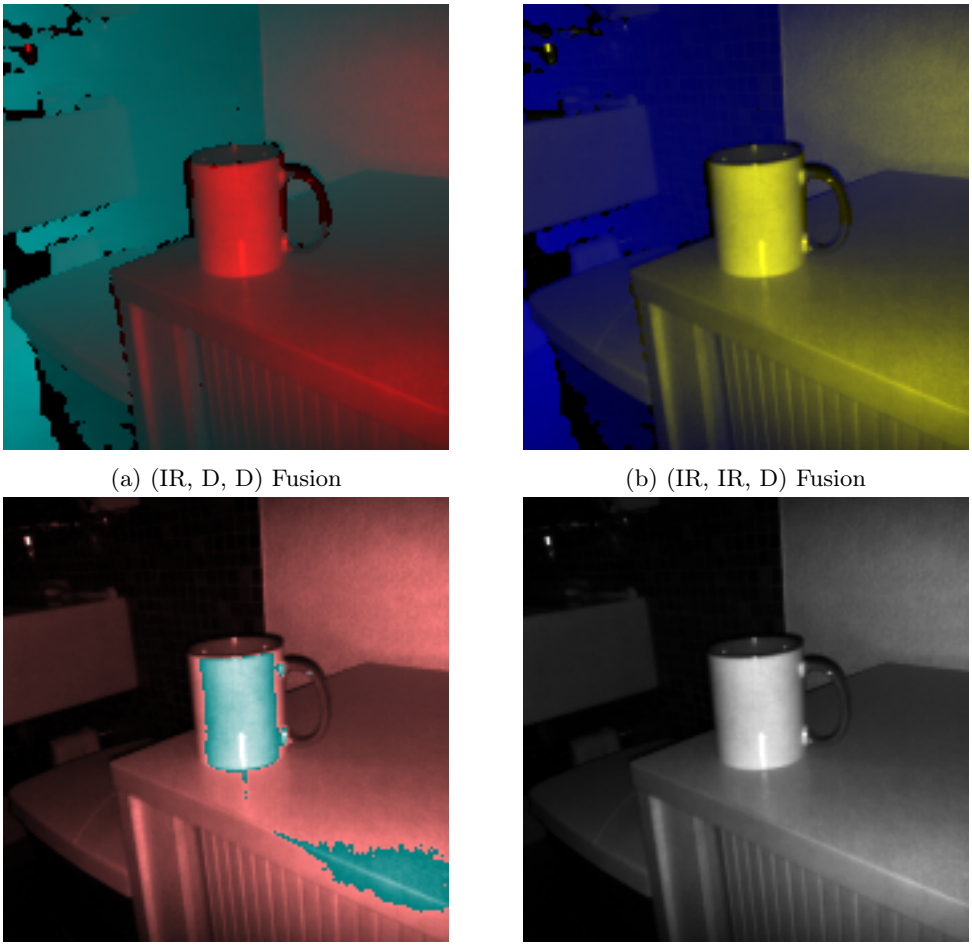
In this fusion strategy, seen in Figure 3.5a, the IR data is combined with two channels of depth data. This approach aims to exploit additional depth information for improved feature representation.

(IR, IR, Depth)

Here, two channels of IR data are fused with one channel of depth data. This configuration, demonstrated in Figure 3.5b, investigates the impact of duplicating the IR modality on the fusion outcome.

(IR, Depth, IR + Depth)

This fusion strategy involves combining IR data with one channel of depth data and a merged representation of both IR and depth data. The combined representation (IR+Depth) aims to capture complementary information from both modalities and can be seen in Figure 3.5c.



(a) (IR, D, D) Fusion

(b) (IR, IR, D) Fusion

(c) (IR, D, IR+D) Fusion

(d) Baseline (IR)

Figure 3.5.: Results of sensor fusion strategies with where the input channels are displayed using RGB color mapping

3.2.3. Data Augmentation

To enhance the robustness and generalization capability of the object detection model, a comprehensive set of data augmentation techniques is employed during the training phase. The used techniques introduce variability and simulate diverse scenarios:

Random Resized Crop

Randomly selects a region of the input image and resizes it to the desired dimension. This augmentation technique increases spatial diversity in the training dataset.

Rotate

Introduces rotational variations to the image, exposing the model to different orientations and viewpoints of the objects.

Random Brightness and Contrast (Random BC)

Adjusts the brightness and contrast levels of the images randomly, contributing to improved adaptability to varying lighting conditions.

Horizontal Flip

Flips the images horizontally, enriching the dataset with horizontally mirrored instances.

Cutout

Randomly masks out square regions of different sizes in the input images. This regularization technique encourages the model to handle possible obstructions of the objects.

Channel Dropout

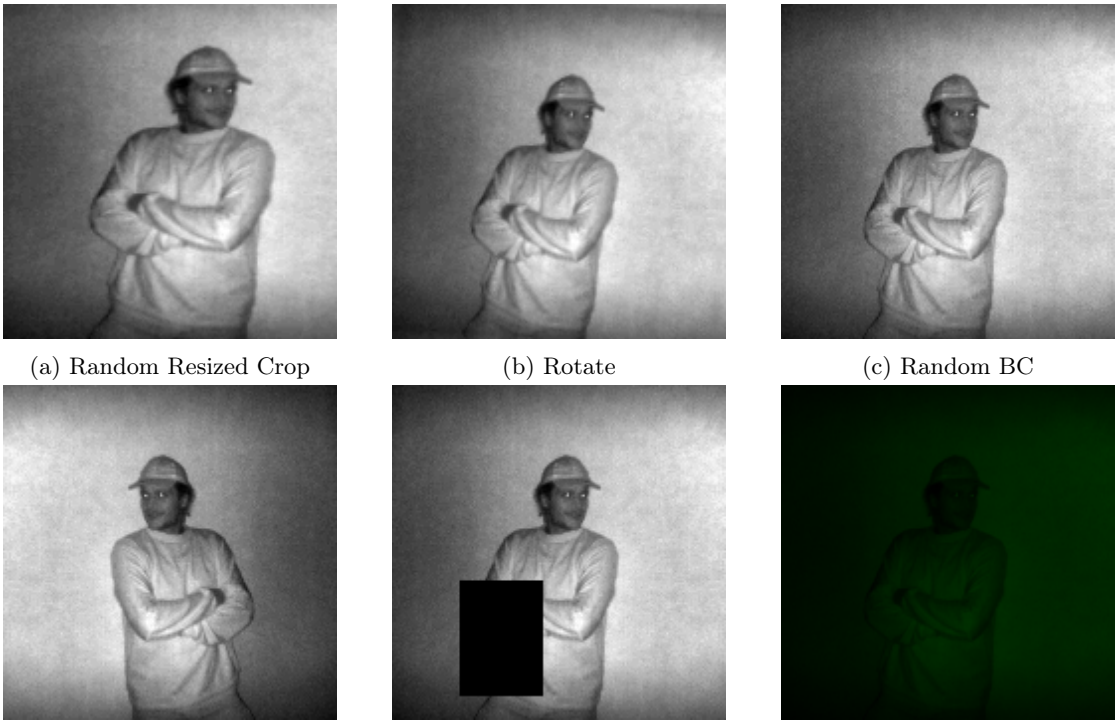
Specifically designed to address potential sensor data loss, Channel Dropout randomly drops out entire channels in the input data. This ensures that the model remains robust even if information from one modality (depth or IR) is partially or entirely missing or useless, relying on the available modality for learning.

3.2.4. Network Architecture

The chosen architecture for this thesis is the FOMO network, specifically designed to operate efficiently on low-power devices, thereby facilitating real-time object detection.

The key concept behind FOMO is to address the object detection challenge through image classification methodologies. Specifically, FOMO adapts the original MobileNetV2 [39] classification architecture, and cuts off the classification head with some supplementary feature extraction layers, shown in Figure 3.7.

Consequently, FOMO employs the identical feature extractor as MobileNetV2 up to a specified cut-off point, where FOMO introduces its own detection head which outputs a per-region class probability map that can be seen in Figure 3.8a. The resolution of this class-probability heat map depends upon the cut-off point in the MobileNetV2 architecture, with an earlier cut-off resulting in a higher resolution. In the FOMO model



(a) Random Resized Crop (b) Rotate (c) Random BC

(d) Horizontal Flip (e) CutOut (f) Channel Dropout

Figure 3.6.: Different data augmentation techniques used, for better visibility only the IR channel was used except in the demonstration in 3.6f, where RGB channels are used to showcase the dropout of the R and B channels

trained for this thesis, the generated heat map is configured to be 8 times smaller than the input image; however, this parameter is adjustable for the specific use case [60].

Furthermore, the generated heat maps are used to predict centroids on the objects instead of the whole bounding boxes, which is the biggest difference from conventional object detection networks. The centroid prediction for an example input image is shown in Figure 3.8b.

3.3. Model Deployment

In the pursuit of deploying complex neural network models on resource-constrained microprocessors, GAPFlow is used, a specialized tool provided by GreenWaves Technologies. GAPFlow expects a trained .onnx or .tflite model file as an input. This trained model can then be quantized to different precisions to reduce the model size, whereas this thesis used int8 quantization. Furthermore, another tool inside GAPFlow, the GAP Autotiler,

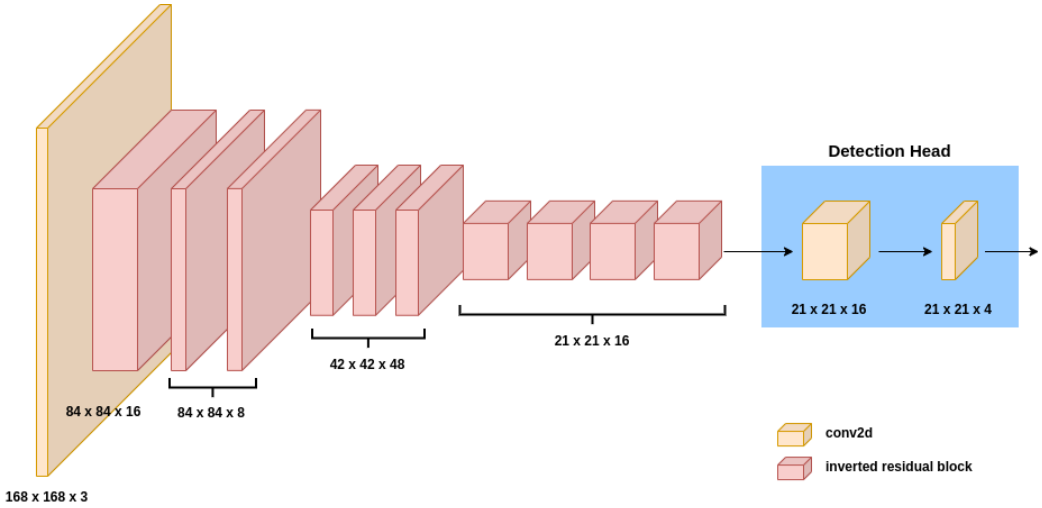
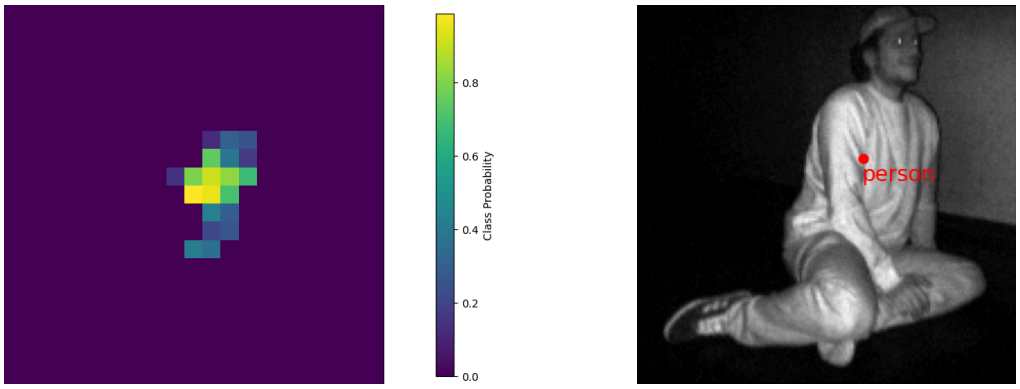


Figure 3.7.: FOMO network architecture



(a) Class-probability heat map for person class

(b) Centroid prediction on input image

Figure 3.8.: Example outputs for FOMO network

is used to automatically create the C code for deployment on microprocessors, specifically targeting the GAP9 architecture.

The GAP9 is a microprocessor that features a 9-core RISC-V compute cluster, an AI accelerator, and a 1-core RISC-V fabric controller. It achieves a power efficiency of 330 µW/GOP processing neural networks, running at a maximum internal clock of 370 MHz [61].

Chapter 4

Results

After presenting the experimental setup for training the object detection model, this section evaluates the performance of the different fusion strategies quantitatively and qualitatively. The trained models are then further used to perform power and performance measurements on the GAP9 microprocessor.

4.1. Training Setup

All training and testing procedures were executed on a laptop featuring an Intel Core i7-1165G7 CPU, ensuring a standardized computational environment throughout the experiments. The neural network employed was the default FOMO network, sourced from Edge Impulse. The configuration of the dataset on the Edge Impulse interface facilitated seamless integration of the FOMO network into the experimental pipeline.

For the training of the FOMO network, a transfer learning approach was employed, utilizing pretrained weights from MobileNetV2 [39]. The MobileNetV2 model was parameterized with a width multiplier of 0.35 and an input resolution of 96x96 pixels. The model pretraining is done on the ImageNet dataset for 260 epochs, employing a batch size of 64. This pretraining process enhances the network's capacity to recognize and extract hierarchical features from input images, thereby improving training for this specific case.

As mentioned in Section 3.1.2, the flexx2 dataset is randomly split into a training and testing set. After performing early fusion on the data, all models are trained for 200 epochs using a batch size of 32 and a learning rate of 0.001, where the weights of best best-performing model are saved. During training, the data augmentation introduced in Section 3.2.3 is applied to counteract overfitting and make the model more robust in challenging situations.

In assessing the performance of the developed object detection model, a comprehensive set of metrics was employed. Notably, the F1 score, which balances precision and recall, was selected as a primary metric for evaluation. Another way of expressing the F1 score is through a rate of true positives (TP), false positives (FP), and false negatives (FN) which can be seen in Equation 4.1.

$$F1 = 2 * \frac{precision * recall}{precision + recall} = \frac{2 * TP}{2 * TP + FP + FN} \tag{4.1}$$

4.2. Fusion Experiments

The exploration of sensor fusion techniques and their impact on object detection performance is a critical aspect of this thesis. The data encapsulated in Table 5.1 provides a comprehensive overview of how different fusion strategies, as discussed in Section 3.2.2, compare against single-channel baselines that utilize either IR or depth information exclusively.

Fusion	**F1 Score**	**Precision**	**Recall**	**#Parameter**
Infrared	0.859	0.85	0.87	**17'916**
Depth	0.805	0.81	0.8	**17'916**
IR, D, D	0.818	0.83	0.8	18'204
IR, IR, D	0.84	0.84	0.84	18'204
IR, D, IR+D	**0.861**	**0.87**	**0.86**	18'204

Table 4.1.: Comparison of different fusion performances

The table illustrates that the hybrid fusion method, which combines IR and depth data in the third channel (IR, D, IR+D), outperforms the other fusion configurations with an F1 Score of 0.861, where the model used a total of 18'204 parameters.

The smallest networks in terms of the number of parameters are both baseline networks, where only a single channel is utilized, resulting in 17'916 network parameters. The single-channel IR model showcases robust performance metrics, with an F1 Score of 0.859, precision of 0.85, and recall of 0.87, while the depth-only model falls short when compared to the IR and other fusion strategies achieving an F1 Score of 0.805. The dual-channel fusions, (IR, D, D) and (IR, IR, D), offer intermediate results, with the latter achieving a balanced precision and recall of 0.84. While the (IR, D, D) fusion records a comparably low recall of 0.8, indicating that it happens to not detect more objects than the other models.

The qualitative analysis presented in Figure 4.1 further substantiates the quantitative findings from Table 5.1. It visually demonstrates the efficacy of each fusion method, where predictions are superimposed onto the IR channel for enhanced clarity. The superior

performance of the (IR, D, IR+D) fusion method is evident when directly contrasting with the predictions from the other techniques. Due to the better performance of the IR baseline against the depth baseline, this model is used for comparison to the other fusion techniques in Figure 4.1 and in the coming sections.

(a) Baseline

(b) (IR, D, D) Fusion

(c) (IR, IR, D) Fusion

(d) (IR, D, IR + D) Fusion

Figure 4.1.: Qualitative results of the different fusion strategies

4.3. Model Performance on GAP9

The C code of the quantized models is further utilized to conduct power measurements and performance assessments of the models. This involves evaluating the inference speed, energy efficiency, the total number of operations required for inference, and the average power consumption. These metrics help in determining the model's performance on

GAP9 and its viability for real-world applications, as well as understanding how different fusion strategies influence the model's power consumption and operational efficiency.

Figure 4.2 depicts a line graph that illustrates the overall power consumption of the GAP9 processor over time, including the power used by its memory components. Initially, there is a noticeable fluctuation in power consumption, characterized by several peaks and troughs. This phase corresponds to the startup period of the GAP9, where the system is awakened and powered on, followed by the initialization of all necessary processing units.

After this initial phase, the power consumption stabilizes and reaches a steady state, which is represented by the more consistent and lower level of power usage towards the right side of the graph. It is at the peak at the beginning of this stable phase that performance measurements for detection inference are recorded.

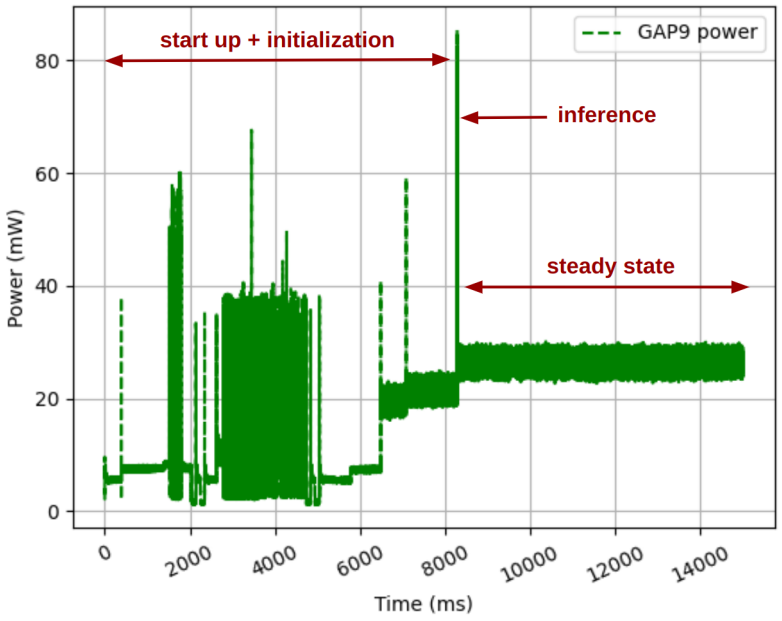
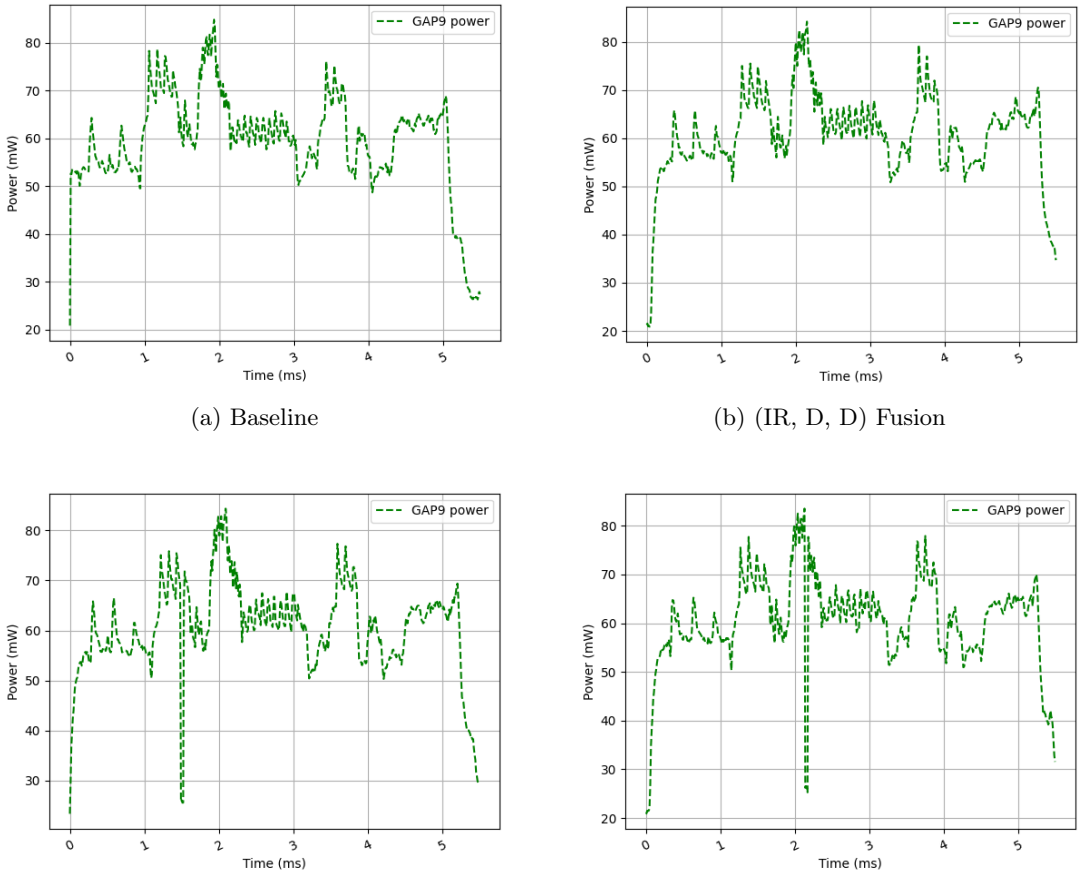


Figure 4.2.: Power measurements on GAP9 over time

Building upon this line graph of the GAP9's performance, the focus goes to the results specific to each fusion strategy time for detection. Figure 4.3 shows a magnified version of the peak in 4.2 for each fusion strategy. The individual graphs provide a visualization of the power consumption profile during a single frame inference when executing different fusion strategies during object detection tasks, where the measurement is started at the initial spike for 5.5 ms before ending. Upon examination, it's observed that the power consumption profiles are quite similar across all strategies. Each graph illustrates an initial surge in power usage after which the power measurements exhibit a series of peaks and valleys, with a maximum consumption of around 85 mW for all the tested models.



(a) Baseline

(b) (IR, D, D) Fusion

(c) (IR, IR, D) Fusion

(d) (IR, D, IR + D) Fusion

Figure 4.3.: Power measurements for inference of the tested fusion approaches

Table 5.1 summarizes the measured performance of the model under the different fusion configurations. The results reveal consistent inference times across all fusion methods, approximately 5.4 ms. The average power consumption for processing a single frame lies around 60 mW for all tested models with an average energy consumption of approximately 330 µJ. The operation count, measured in MOps, was slightly higher for the fusion strategies compared to the infrared-only baseline, due to inputting three instead of one channel into the network.

Fusion	Inference Time	MOps	Average Power	Energy
Infrared	~5.4 ms	16.03	59.45 mW	326.98 µJ
IR, D, D	~5.4 ms	18.03	60.77 mW	347.62 µJ
IR, IR, D	~5.4 ms	18.03	60.61 mW	333.35 µJ
IR, D, IR+D	~5.4 ms	18.03	60.47 mW	332.62 µJ

Table 4.2.: Comparison of Fusion Performances on GAP9

Chapter 5

Discussion

This section explores the reasons behind the lack of improvement in performance metrics upon integrating depth data and provides a qualitative assessment of the benefits that depth information brings to the networks.

5.1. Impact of Depth in Detection

Surprisingly, the introduction of depth in early fusion did not yield a discernible increase in the F1 score, as seen in Table 5.1. This unexpected observation challenges the initial hypothesis that the additional information from these modalities would inherently enhance the model's performance. However, when examining the addition of IR and depth in the third channel an increase in F1 score emerged. While the increase in F1 score was marginal and statistically non-significant, a closer inspection of test predictions revealed noteworthy nuances that can already be seen in the high confidence predictions in Figure 4.1d. Notably, in scenarios characterized by increased distance between the camera and the object, the fusion of depth information subtly amplified the network's proficiency.

The absence of a statistically significant improvement in the F1 score may be attributed to the dominance of data where depth information did not confer a substantial advantage. Nevertheless, the qualitative analysis of test predictions unveiled instances where the fusion of additional depth exhibited a perceptible impact, particularly in challenging scenarios where the target object was positioned at a considerable distance, which can be seen in Figure 5.1c.

This finding suggests that while the used performance metrics may not reflect a substantial enhancement, the fusion of depth information in specific challenging cases has the potential to refine the model's predictions.

(a) Depth image recorded from flexx2

(b) Predicition with baseline network

(c) Prediction fusing additional depth information

(d) Depth image recorded from flexx2

(e) Predicition with baseline network

(f) Prediction fusing additional depth information

Figure 5.1.: Examples of positive effect of fusing depth data

5.2. Impact of Depth on GAP9 Performance

While other work [2] measured power on GAP9 over multiple consecutive frames using a small detection network. This thesis only evaluated power measurement for processing a single input frame performed for each model, which makes statistically interpreting the results difficult. The results still give us insight into some aspects that can be drawn from the measurements.

The uniform inference time across all fusion methods (~5.4 ms) indicates that the integration of additional data streams does not adversely affect the processing speed in this specific case, maintaining a consistent temporal performance across different configurations.

The marginal differences in average power consumption among the various configurations highlight the energy-efficient nature of the early fusion approach. Despite the inclusion of depth data, the increase in power requirement is minimal, demonstrating the feasibility of implementing such fusion techniques in power-sensitive applications.

The experiments in Section 4.3 reveal that the integration of depth data into object detection systems on the GAP9 platform is characterized by a negligible impact on inference speed and a modest increase in computational and power demands.

Comparing the made power measurements with the smallest and the best-performing network from [2] brings into perspective how low-power and fast the trained networks are. The best-performing model in this thesis is predicting 3x faster while having 22x fewer parameters and using 34 mW less power for detecting a single frame compared to the smallest model from [2].

Network	**Inference Time**	**Parameters**	**Average Power**	**Energy**
TYv8 big [2]	34 ms	839k	-	2.62 mJ
TYv1.3 small [2]	16.86 ms	403k	94.10 mW	1.59 mJ
Ours	5.4 ms	18.2k	60.47 mW	0.33 mJ

Table 5.1.: Comparison of fusion performances on GAP9

Chapter 6

Conclusion and Future Work

This thesis presents a novel dataset tailored specifically for object detection tasks, comprising 1412 IR and depth image pairs. This unique dataset enabled us to explore and evaluate various data fusion approaches for object detection.

Furthermore, the FOMO [12] model is used for sensor fusion of the data recorded with the flexx2 camera. FOMO stands out by delivering rapid detection speeds alongside significantly reduced power consumption, showcasing its superiority over state-of-the-art lightweight detection networks that are often larger and require higher computational demands. This makes FOMO particularly advantageous for deployment in real-world applications where resources are limited and efficiency is paramount, which is underlined by the performance measurements on the GAP9 microprocessor.

While the quantitative results did not show a significant benefit in fusing depth data to the IR input, the qualitative evaluation underscores the potential of combining IR and depth data for enhancing object detection capabilities, especially in adverse light conditions. Notably, the utilization of the flexx2 camera played a pivotal role in the work, empowering the object detection framework to maintain good performance even under challenging lighting conditions where traditional cameras falter. The flexx2 camera also enables the extraction of depth information from detected objects, which could be interesting for obstacle avoidance or tracking in further research.

In the future, an avenue for exploration also involves assessing the real-time performance of the GAP9 microprocessor in conjunction with live data from the flexx2 camera, particularly with our FOMO implementation in operation. Although the performance measurements showed the compatibility of the trained model with GAP9 for real-time execution, challenges associated with the flexx2 camera's interface did not allow direct live inference testing. A crucial element that could facilitate this future work is the common CSI-2 (Camera Serial Interface-2) interface shared by the flexx2 camera's sensor and the GAP9 microprocessor. This shared interface not only underpins the technical feasibility of our

envisioned deployment but also promises to simplify the integration process, paving the way for seamless live data processing and inference. Advancing this work could significantly contribute to the practical application of real-time object detection systems, leveraging the strengths of both the flexx2 camera's advanced sensing capabilities and the GAP9's efficient processing power.

Appendix A

Task Description

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

CENTER FOR PROJECT-BASED LEARNING

Task Description for a Semester Thesis on

Sensor Fusion with RGB and Depth Cameras on GAP9

at the Department of Information Technology and Electrical Engineering

for

Jonas Bohn
bohnj@student.ethz.ch

Advisors:	Hanna Müller, hanmuell@iis.ee.ethz.ch Tommaso Polonelli, topolonelli@pbl.ee.ethz.ch
Professor:	Prof. Dr. Luca Benini, lbenini@ethz.ch
Handout Date:	23.10.2021
Due Date:	29.01.2022

Project Goals

Percepting the environment is an important task especially in robotics. However, often the resource onboard a mobile robot are limited but should still work in various environmental conditions. Weight and power constraints limit the types of sensors that can be mounted as well as the available processing power. Visual pipelines for autonomous navigation have been deployed on sub-100mW multi-core system-on-chips (SoCs) before, as [5][6], however they are not capable to function in dark environments. To improve the accuracy of a system, several low-power low-resolution sensors can be combined in a so called sensor fusion [3, 4]. Usually those algorithms are executed on powerful computers, however sometimes the power budget and space are limited. To bring a neural network to a resource constraint embedded system, the network is first quantized to save space (e.g. to 8-bits for weights and activations) and then deployed using specialized tools as TFLite, DORY [2] or the NNTool.

The target for the deployment is a risc-v-based multi-core processor, as GAP9 from greenwaves - it is a general purpose microcontroller with a NE16 hardware accelerator and a 9-core cluster and therefore can handle various network structures.

In this thesis we want to target sensor fusion with grayscale and depth cameras on a resource-constrained platform. As sensors we will use a flexx2 camera - this thesis will explore if fusing depth and visual information at high resolution can improve reliability of object detection.

Tasks

The project will be split into three phases, as described below:

Phase 1 (Week 1-4)

1. Investigate the state-of-the-art of the sensor fusion of depth and visual information for object detection
2. Get used to the hardware, i.e. learn how to acquire data with the sensor and/or in simulation, get familiar with available tools for training, quantization and deployment
3. Get familiar with previous work, as the TinyssimoYOLO and the YOLOv5p ported to GAP9.

Phase 2 (Week 5-11)

1. Continuing the data acquisition and consolidating the machine learning part.
2. Quantization and deployment of a selected model on the GAP9 microcontroller.
3. Test, evaluation, and characterization of the deployed model.

4. (bonus) Live demo - sending the data from the camera to the GAP9 over UART (or similar) and doing life inference.

Phase 3 (Week 12-14)

1. Performance (power, latency) measurements on GAP9
2. Report writing and presentation preparation

Milestones

The following milestones need to be reached during the thesis:

- Testing and characterization of the depth sensor and dataset acquired (or from previous works).
- Choosing and evaluating an ML model for sensor fusion.
- Porting of the model to GAP9.
- Performance measurements.
- Final report and presentation.

Project Organization and Grading

During the thesis, students will gain experience in the independent solution of a technical-scientific problem by applying the acquired specialist and social skills.

The grade is based on the following: Student effort; thoroughness and learning curve; achieving qualitative and quantitative results with a scientific approach; supporting practical findings with theoretical background and literature investigations; final presentation and report; documentation and reproducibility. All theses include an oral presentation, a written report and are graded. The report and presentation need to have publication grade quality to achieve a good grade. Students are graded based on the official ITET grading form[1].

For students of IIS (Prof. Benini) a special grading scheme exists, please contact your supervisor for details there. Before starting, the project must be registered in myStudies and all required documents need to be handed in for archiving by PBL.

Laboratory Rules

The students agree to follow the lab rules set by PBL staff, for detail please contact us. The most important points are:

[1] https://ethz.ch/content/dam/ethz/special-interest/itet/department/Studies/Forms/Grading%20Form.xlsx

- All ETH safety regulations need to be followed[2], in addition to ones given by PBL staff
- No device in the lab is used without introduction by your supervisor or PBL staff
- No device leaves the lab without being offically borrowed, this is done by PBL staff and needs your Legi.
- Any damage to devices or tools needs to be reported immediately to PBL staff.
- The Lab-desk is clean and free for others after you finished your task, or when you take longer breaks. All tools are correctly sorted into their drawers/cupboards when you leave

Weekly Report

There will be a weekly report/meeting held between the student and the assistants. The exact time and location of these meetings will be determined within the first week of the project in order to fit the students and the assistants schedule. These meetings will be used to evaluate the status and document the progress of the project (required to be done by the student). Beside these regular meetings, additional meetings can be organized to address urgent issues as well. The weekly report, along with all other relevant documents (source code, datasheets, papers, etc), should be uploaded to a clouding service, such as Polybox and shared with the assistants.

Project Plan

Within the first month of the project, you will be asked to prepare a project plan. This plan should identify the tasks to be performed during the project and sets deadlines for those tasks. The prepared plan will be a topic of discussion of the first week's meeting between you and your assistants. Note that the project plan should be updated constantly depending on the project's status.

Final Report and Paper

PDF copies of the final report written in English are to be turned in. Basic references will be provided by the supervisors by mail and at the meetings during the whole project, but the students are expected to add a considerable amount of their own literature research to the project ("state of the art").

[2] https://ethz.ch/staffnet/en/service/safety-security-health-environment/sicherheit-in-laboren-und-werkstaetten/laborsicherheit.html

Final Presentation

There will be a presentation (15 min presentation and 5 min Q&A for BT/ST and 20 min presentation and 10 min Q&A for MT) at the end of this project in order to present your results to a wider audience. The exact date will be determined towards the end of the work.

References

Will be provided by the supervisors by mail and at the meetings during the whole project.

Place and Date Zurich, 12.10.23 Signature Student 

Bibliography

[1] Conti, Francesco, Technical Report: NEMO DNN Quantization for Deployment Model, preprint 2020

[2] A.Burello et al., DORY: Automatic End-to-End Deployment of Real-World DNNs on Low-Cost IoT MCUs, 2021 IEEE Transactions on Computers

[3] Ophoff, Tanguy and Van Beeck, Kristof and Goedemé, Toon. (2019). Exploring RGB+Depth Fusion for Real-Time Object Detection. Sensors. 19. 866. 10.3390/s19040866.

[4] Arif Maliha, Yong Calvin, Mahalanobis Abhijit. (2022). Background Invariant Classification on Infrared Imagery by Data Efficient Training and Reducing Bias in CNNs.

[5] Imagery by Data Efficient Training and Reducing Bias in CNNs. V. Niculescu, L. Lamberti, F. Conti, L. Benini and D. Palossi, "Improving Autonomous Nano-Drones Performance via Automated End-to-End Optimization and Deployment of DNNs," in IEEE Journal on Emerging and Selected Topics in Circuits and Systems, vol. 11, no. 4, pp. 548-562, Dec. 2021, doi: 10.1109/JETCAS.2021.3126259.

[6] D. Palossi et al., "Fully Onboard AI-Powered Human-Drone Pose Estimation on Ultralow-Power Autonomous Flying Nano-UAVs," in IEEE Internet of Things Journal, vol. 9, no. 3, pp. 1913-1929, 1 Feb.1, 2022, doi: 10.1109/JIOT.2021.3091643.

[7] V. Niculescu, H. Müller, I. Ostovar, T. Polonelli, M. Magno and L. Benini, "Towards a Multi-Pixel Time-of-Flight Indoor Navigation System for Nano-Drone Applications," 2022 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), 2022, pp. 1-6, doi: 10.1109/I2MTC48687.2022.9806701.

Appendix B

Declaration of Originality

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

Clear as Day: Low-Power Object Detection for Challenging Conditions

Authored by (in block letters):

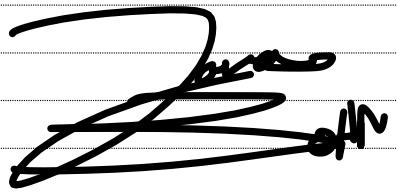
For papers written by groups the names of all authors are required.

Name(s):	**First name(s):**
Bohn	Jonas

With my signature I confirm that

- I have committed none of the forms of plagiarism described in the 'Citation etiquette' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date	**Signature(s)**
Zurich, 3.2.2024	

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.

Bibliography

[1] J. Redmon and A. Farhadi, "Yolo9000: Better, faster, stronger," 2016.

[2] J. Moosmann, P. Bonazzi, Y. Li, S. Bian, P. Mayer, L. Benini, and M. Magno, "Ultra-efficient on-device object detection on ai-integrated smart glasses with tinyissimoyolo," 2023.

[3] pmdtechnologies ag, *flexx2 / Getting Started Guide*, 2022.

[4] A. Gupta, A. Anpalagan, L. Guan, and A. S. Khwaja, "Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues," *Array*, vol. 10, p. 100057, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2590005621000059

[5] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," *Neural Information Processing Systems*, vol. 25, 01 2012.

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[7] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[8] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, "Microsoft coco: Common objects in context," 2015.

[9] V. Kamath and A. Renuka, "Deep learning based object detection for resource constrained devices: Systematic review, future trends and challenges ahead," *Neurocomputing*, vol. 531, pp. 34–60, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231223001388

[10] M. Bijelic, T. Gruber, F. Mannan, F. Kraus, W. Ritter, K. Dietmayer, and F. Heide, "Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather," 2020.

[11] C. Brander, C. Cioflan, V. Niculescu, H. Müller, T. Polonelli, M. Magno, and L. Benini, "Improving data-scarce image classification through multimodal synthetic data pretraining," in *2023 IEEE Sensors Applications Symposium (SAS)*, 2023, pp. 1–6.

[12] L. Moreau and M. Kelcey, "Announcing fomo (faster objects more objects)," *Edge Impulse*, 2022, [Online; accessed 18-January-2024]. [Online]. Available: https://www.edgeimpulse.com/blog/announcing-fomo-faster-objects-more-objects/

[13] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," 2023.

[14] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," 2014.

[15] ——, "Region-based convolutional networks for accurate object detection and segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 1, pp. 142–158, 2016.

[16] R. Girshick, "Fast r-cnn," 2015.

[17] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," 2016.

[18] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, *SSD: Single Shot MultiBox Detector*. Springer International Publishing, 2016, p. 21–37. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-46448-0_2

[19] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[20] H.-T. Pham, M.-A. Nguyen, and C.-C. Sun, "Aiot solution survey and comparison in machine learning on low-cost microcontroller," in *2019 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, 2019, pp. 1–2.

[21] S. Han, J. Pool, J. Tran, and W. J. Dally, "Learning both weights and connections for efficient neural networks," 2015.

[22] S. Swaminathan, D. Garg, R. Kannan, and F. Andres, "Sparse low rank factorization for deep neural network compression," *Neurocomputing*, vol. 398, pp. 185–196, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231220302253

[23] A. Gholami, S. Kim, Z. Dong, Z. Yao, M. W. Mahoney, and K. Keutzer, "A survey of quantization methods for efficient neural network inference," 2021.

[24] Y. Gong, L. Liu, M. Yang, and L. D. Bourdev, "Compressing deep convolutional networks using vector quantization," *CoRR*, vol. abs/1412.6115, 2014. [Online]. Available: http://arxiv.org/abs/1412.6115

[25] D. D. Lin, S. S. Talathi, and V. S. Annapureddy, "Fixed point quantization of deep convolutional networks," 2016.

[26] R. Li, Y. Wang, F. Liang, H. Qin, J. Yan, and R. Fan, "Fully quantized network for object detection," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 2805–2814.

[27] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," 2016.

[28] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, "Pruning convolutional neural networks for resource efficient transfer learning," *ArXiv*, vol. abs/1611.06440, 2016. [Online]. Available: https://api.semanticscholar.org/CorpusID:16167970

[29] D. Blalock, J. J. G. Ortiz, J. Frankle, and J. Guttag, "What is the state of neural network pruning?" 2020.

[30] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015.

[31] A. Polino, R. Pascanu, and D. Alistarh, "Model compression via distillation and quantization," 2018.

[32] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," 2017.

[33] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Q. V. Le, and H. Adam, "Searching for mobilenetv3," 2019.

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

[35] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," 2020.

[36] J. Moosmann, M. Giordano, C. Vogt, and M. Magno, "Tinyissimoyolo: A quantized, low-memory footprint, tinyml object detection network for low power microcontrollers," in *2023 IEEE 5th International Conference on Artificial Intelligence Circuits and Systems (AICAS)*. IEEE, Jun. 2023. [Online]. Available: http://dx.doi.org/10.1109/AICAS57966.2023.10168657

[37] N. Zimmerman, H. Müller, M. Magno, and L. Benini, "Fully onboard low-power localization with semantic sensor fusion on a nano-uav using floor plans," 2023.

[38] B. Wu, A. Wan, F. Iandola, P. H. Jin, and K. Keutzer, "Squeezedet: Unified, small, low power fully convolutional neural networks for real-time object detection for autonomous driving," 2019.

[39] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," 2019.

[40] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," 2020.

[41] R. J. Wang, X. Li, and C. X. Ling, "Pelee: A real-time object detection system on mobile devices," 2019.

[42] S. Chen, T. Cheng, J. Fang, Q. Zhang, Y. Li, W. Liu, and X. Wang, "Tinydet: Accurate small object detection in lightweight generic detectors," 2023.

[43] Z. Qin, Z. Li, Z. Zhang, Y. Bao, G. Yu, Y. Peng, and J. Sun, "Thundernet: Towards real-time generic object detection," 2022.

[44] C. Shiqi, Z. Ronghui, W. Wang, and J. Yang, "Learning slimming sar ship object detector through network pruning and knowledge distillation," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. PP, pp. 1–1, 12 2020.

[45] L. Zhong, H. Tan, and J. Liao, "Yolox-nano: Intelligent and efficient dish recognition system," in *2022 7th International Conference on Intelligent Computing and Signal Processing (ICSP)*, 2022, pp. 1391–1395.

[46] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLO," Jan. 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[47] F. Farahnakian and J. Heikkonen, "A comparative study of deep learning-based rgb-depth fusion methods for object detection," in *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2020, pp. 1475–1482.

[48] ——, "Deep learning based multi-modal fusion architectures for maritime vessel detection," *Remote Sensing*, vol. 12, p. 2509, 08 2020.

[49] T. Ophoff, K. Van Beeck, and T. Goedemé, "Exploring rgb+depth fusion for real-time object detection," *Sensors*, vol. 19, no. 4, 2019. [Online]. Available: https://www.mdpi.com/1424-8220/19/4/866

[50] F. Farahnakian, J. Poikonen, M. Laurinen, D. Makris, and J. Heikkonen, "Visible and infrared image fusion framework based on retinanet for marine environment," in *2019 22th International Conference on Information Fusion (FUSION)*, 2019, pp. 1–7.

[51] S. Vora, A. H. Lang, B. Helou, and O. Beijbom, "Pointpainting: Sequential fusion for 3d object detection," 2020.

[52] H. Gao, B. Cheng, J. Wang, K. Li, J. Zhao, and D. Li, "Object classification using cnn-based fusion of vision and lidar in autonomous vehicle environment," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 4224–4231, 2018.

[53] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3354–3361.

[54] K. Liu, Y. Li, N. Xu, and P. Natarajan, "Learn to combine modalities in multimodal deep learning," 2018.

[55] D. Xu, D. Anguelov, and A. Jain, "Pointfusion: Deep sensor fusion for 3d bounding box estimation," 2018.

[56] T. Zhou, D.-P. Fan, M.-M. Cheng, J. Shen, and L. Shao, "Rgb-d salient object detection: A survey," *Computational Visual Media*, vol. 7, no. 1, p. 37–69, Jan. 2021. [Online]. Available: http://dx.doi.org/10.1007/s41095-020-0199-z

[57] L. Crupi, E. Cereda, A. Giusti, and D. Palossi, "Sim-to-real vision-depth fusion cnns for robust pose estimation aboard autonomous nano-quadcopters," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 7711–7717.

[58] L. Lamberti, L. Bompani, V. J. Kartsch, M. Rusci, D. Palossi, and L. Benini, "Bio-inspired autonomous exploration policies with cnn-based object detection on nano-drones," 2023.

[59] E. Impulse, "Labeling queue (object detection)," 2023, [Online; accessed 22-January-2024]. [Online]. Available: https://docs.edgeimpulse.com/docs/edge-impulse-studio/data-acquisition/labeling-queue

[60] ——, "Fomo: Object detection for constrained devices," 2023, [Online; accessed 23-January-2024]. [Online]. Available: https://docs.edgeimpulse.com/docs/edge-impulse-studio/learning-blocks/object-detection/fomo-object-detection-for-constrained-devices

[61] greenwaves technologies, "Gap9 product brief," 2021, [Online; accessed 26-January-2024]. [Online]. Available: https://greenwaves-technologies.com/wp-content/uploads/2023/02/GAP9-Product-Brief-V1_14_non_NDA.pdf

[62] F. Nobis, M. Geisslinger, M. Weber, J. Betz, and M. Lienkamp, "A deep learning-based radar and camera sensor fusion architecture for object detection," 2020.

[63] pmdtechnologies ag, "flexx2 3d camera data brief," 2024, [Online; accessed 22-January-2024]. [Online]. Available: https://3d.pmdtec.com/en/3d-cameras/flexx2/

[64] L. Yijin, L. Xinyang, D. Wenqi, Z. han, B. Hujun, Z. Guofeng, Z. Yinda, and C. Zhaopeng, "Deltar: Depth estimation from a light-weight tof sensor and rgb image," in *European Conference on Computer Vision (ECCV)*, 2022.

[65] G. Mora-Martín, A. Turpin, A. Ruget, A. Halimi, R. Henderson, J. Leach, and I. Gyongy, "High-speed object detection with a single-photon time-of-flight image sensor," *Optics Express*, vol. 29, no. 21, p. 33184, Sep. 2021. [Online]. Available: http://dx.doi.org/10.1364/OE.435619