# Exploiting Scene Depth for Object Detection with Multimodal Transformers

Hwanjun Song[1,*], Eunyoung Kim[2]
Varun Jampani[2], Deqing Sun[2]
Jae-Gil Lee[3], Ming-Hsuan Yang[2,4,5]

[1]NAVER AI Lab, [2]Google Research
[3]KAIST, [4]University of California Merced
[5]Yonsei University

**Abstract**

We propose a generic framework MEDUSA (Multimodal Estimated-Depth Unification with Self-Attention) to fuse RGB and depth information using multimodal transformers in the context of object detection. Unlike previous methods that use the depth measured from various physical sensors such as Kinect and Lidar, we show that the depth maps inferred by a monocular depth estimator can play an important role to enhance the performance of modern object detectors. In order to make use of the estimated depth, MEDUSA encompasses a robust feature extraction phase, followed by multimodal transformers for RGB-D fusion. The main strength of MEDUSA lies in its broad applicability for any existing large-scale RGB datasets including PASCAL VOC and Microsoft COCO. Extensive experiments with three datasets show that MEDUSA achieves higher precision than several strong baselines.

## 1 Introduction

The advances of deep neural networks (DNNs) have expedited the development of accurate object detection methods, such as Faster-RCNN [30], SSD [17], YOLO [29], and DETR [2]. However, most existing methods still struggle with close (or overlapping) objects, complex background, and varying illumination [4, 10]. These methods rely on only color intensity for detecting multiple objects and may not recognize geometric variations of the objects in such subtle situations. To complement the inherent limitation of color information, one could also use *depth* information as it is less sensitive to color or lighting variations [4, 36, 45].

Significant performance improvements have been witnessed by exploiting depth information in scene classification [14, 44] and semantic segmentation [36, 47]. With this increasing interest in depth information, numerous RGB-D datasets, such as NYU [32], KITTI [20], and SIP [4], have been constructed. However, depth information is available for a very small proportion of images because acquiring the ground-truth depth at scale still remains a challenge, and thus its widespread adoption in object detection is hindered.

To alleviate the lack of depth information, leveraging estimated depth maps using off-the-shelf models is a promising direction for object detection as recent monocular depth estimators such as [1, 15, 24, 27] are shown to perform quite well and robust across different scene types. For instance, MiDaS [27] can effectively learn to infer dense depth maps from a single-view image by mixing multiple available RGB-D datasets during training. The

* The work was conducted during doing an internship at Google Research.

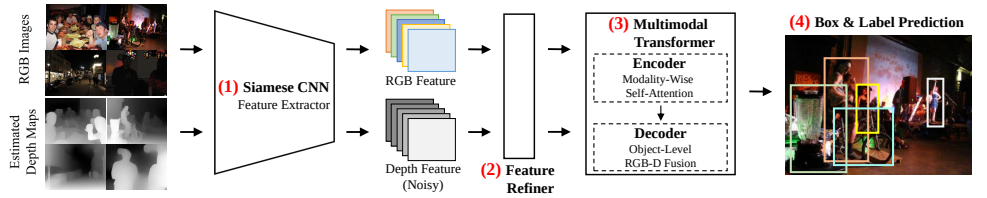



Figure 1: **Overview of the proposed framework**: MEDUSA encompasses a robust feature extractor for RGB and estimated depth inputs, followed by the multimodal transformer for determining their complementary fusion via the attention mechanism.

robustness and generality of recent estimators facilitate numerous vision tasks leveraging depth maps from regular RGB images *without* requiring the ground-truth depth information.

In this paper, we propose a generic object detection framework, **MEDUSA**, that can leverage inferred depth maps with RGB images with multimodal transformers. Differently from the typical early and late fusion schemes for RGB-D input [7, 22, 41, 42], our detection framework consists of the three components shown in Figure 1:

- The **Siamese network** learns hierarchical features from RGB and depth inputs through shared parameters. By the transferability of deep neural networks, each single-channel depth map is duplicated into three channels and treated as a grayscale image to employ the same backbone as for RGB images. This conversion relieves the requirement of a suitable pretrained backbone for *every* modality.
- The **feature refiner** further processes the extracted features. To minimize the negative effects from erroneous depth information, we use *region-* and *channel-wise* attention to robustly focus on reliable regions and channels. We guide the region-wise attention by using a global saliency map of RGB features to be robust to noisy depth measurements.
- The **multimodal transformer** first reasons the RGB and depth features that can contribute to finding objects via self-attention in the encoder. This process helps simplify object extraction and localization for the decoder. Then, the decoder performs *object-level* RGB-D fusion, which applies different fusion strategies per object. Given a fixed number of learned object queries, MEDUSA captures complex correlations between RGB and depth features for each object and directly outputs the final set of predictions.

To our knowledge, this is the *first* generic framework that can leverage depth for traditional object detection with multimodal transformers. Compared to most previous work on early and late fusion that simply concatenates RGB and depth inputs (or features), MEDUSA consists of three well-designed components that work well even with the inferred depth. In particular, MEDUSA determines a fusion strategy for each object individually by employing the transformer architecture based on the DETR detector [2]. Our framework is flexible such that each component can be easily replaced with a better one when available; the performance benefit of leveraging our framework remains consistent even when using the most recent detector, Deformable-DETR [48], as a wheel to build our multi-modal transformers. We evaluate our method on three benchmark datasets with inferred or ground truth depth maps: Pascal VOC [8], Microsoft COCO [16], and SUN-RGBD [33]. MEDUSA outperforms existing early and late fusion approaches for object detection. More precisely, each component of MEDUSA contributes to improving the performance in a synergistic manner.

## 2 Related Work

**RGB-D Detection** Learning rich features from the combination of RGB and depth information has attracted much attention in recent years. In particular, semantic segmen-

| Domain | Method | Depth Format | Feature Extractor | Object Detector | Dataset |
|---|---|---|---|---|---|
| Indoor | Gupta et al. [9] | HHA | Two-stream | SVM | NYUv2 |
| | Hou et al. [10] | HHA | Two-stream | R-CNN | NYUv2 |
| | Xu et al. [31] | HHA | Three-stream | Fast-RCNN | SUN RGB-D |
| | Macanu et al. [21] | Depth Map | Two-stream | Faster-RCNN | SUN RGB-D |
| | Ophoff et al. [23] | HHA | Two-stream | R-CNN | CENTAURO |
| Car Driving | Yang et al. [42] | Depth Map | Two-stream | YOLOv2 | KITTI |
| | Schwarz et al. [30] | Depth Map | Two-stream | Faster-RCNN | Private |
| **Various** | **MEDUSA** | **Gray Image** | **One-stream** | **DETR** | **VOC & COCO** |

Table 1: **Comparison of recent RGB-D object detection methods**: each method is grouped into their target domain. In the first row, "Depth Format" is the encoding for the depth input; "Feature Extractor" is # backbone streams for the RGB and depth inputs; "Object Detector" is the architecture used for detection; and "Dataset" is the dataset used for training.

tation [11, 23, 47] and salient object detection [5, 6, 19] have been actively studied and achieved significant performance improvements. Most approaches have focused on pixel-wise categorization and attempted to overcome the limitation of the RGB input by additionally using depth information. On the other hand, RGB-D object detection is yet to be widely studied owing to the lack of large-scale RGB-D datasets.

Such deficiency in RGB-D datasets motivates recent efforts on specific domains such as indoor scenarios [9, 10, 21, 31] and car driving scenarios [22, 42]. Table 1 summarizes recently proposed RGB-D object detection methods, including MEDUSA which has the following properties:

1. The ground-truth depth map or HHA[1] [7] is *no longer* required and is replaced by estimated depth. Since the feature refiner helps use erroneous depth maps, our method can be easily applied to other datasets that include multiple different domains, such as Pascal VOC [3] and Microsoft COCO [16].
2. The issue of increased model complexity is alleviated by employing a *Siamese* network as the *shared* feature extractor for RGB and depth inputs. In contrast, most of the recent methods maintain *two* independent backbones (i.e., one for each modality), thereby almost doubling the number of parameters for the extractor.
3. The RGB and depth features are fused by the attention mechanism *object-wisely* in the transformers, as opposed to the late fusion scheme in the recent methods that simply concatenate the entire feature map at once at the end of their backbone streams.

**Depth Estimation** The main challenge in monocular depth estimation is to cover a variety of scenarios for its practicality [12, 25]. To this end, numerous learning-based estimators, such as MegaDepth [15] and MiDaS [27], exploit multiple RGB-D datasets with distinct characteristics and biases. As a result, the robustness and generality of the estimators have significantly improved, reaching the weighted human disagreement rate of 12.27%.

**Multimodal Transformer** Multimodal transformers are attracting great attention in the field of multimodal language analysis [26, 43], and self-attention has been successfully applied to various problems including vision-to-language alignment [34]. In this work, we adopt the multimodal transformer to determine the complementary fusion strategy of RGB and depth features for object detection.

**Detection with Inferred Depth Maps** A few methods that utilize depth maps inferred from RGB inputs to help 3D object detection [37, 38] have been proposed in recent years. Pseudo-LiDar [38] uses per-pixel camera position information derived from the depth map, but it still

[1] HHA is a depth encoding consisting of horizontal disparity, height above ground, and norm angle.

requires a pair of left-right images as the reference for detection. In contrast, ForeSeE [37] estimates more precise depth maps by treating foreground and background differently for a better 3D object detection. Different from the existing work, we study how to exploit noisy (inferred) depth maps for the traditional 2D object detection with help of a more effective RGB-D fusion approach based on the transformer.

# 3 MEDUSA Framework

We describe the overall architecture of the proposed framework illustrated in Figure 1, where the three components are designed for **(i)** feature extraction from RGB and depth inputs, **(ii)** feature refinement of RGB and depth features, and **(iii)** feature fusion between them.

## 3.1 Siamese Feature Extractor

A *Siamese* structure (shared backbone) is commonly adopted to extract discriminative features between two different inputs with triplet or contrastive losses [39]. Another property of the Siamese network is its *transferability* to learn hierarchical features from different modalities [6, 13]. In this work, we convert a single-channel depth map to a *three-channel grayscale* image by replicating its single channel map, thereby sharing the Siamese ResNet-50 [2] [8] backbone pretrained on the ImageNet dataset. For each image, our feature extractor receives a pair of an RGB image and a three-channel grayscale image, $X^{rgb} \in \mathbb{R}^{h_0 \times w_0 \times 3}$ and $X^{depth} \in \mathbb{R}^{h_0 \times w_0 \times 3}$, and then generates a feature map for each modality, $F^{rgb} \in \mathbb{R}^{h \times w \times c}$ and $F^{depth} \in \mathbb{R}^{h \times w \times c}$, where $c$ is set to 2048, and $h$ and $w$ are set to $\frac{h_0}{32}$ and $\frac{w_0}{32}$.

## 3.2 Feature Refiner

Although a monocular depth estimator infers a depth map from a monocular image fairly well, the depth map inevitably contains non-negligible *estimation errors*; specifically, it performs well for moderately large or close objects, but not for small or distant objects.

For example in Figure 2, smaller birds and farther cars are missed in the depth maps, indicating depth features are incorrectly generated from these erroneous regions. To enhance the robustness to noisy depth maps, our feature refiner adopts the idea of *region-* and *channel-wise* attentions to reweight the feature maps obtained from the Siamese network.

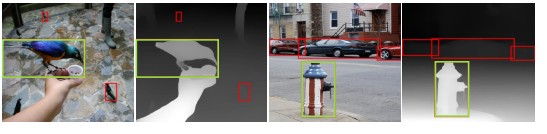


Figure 2: Examples of incorrectly estimated depth maps by MiDas [27] with their RGB images and ground-truth bounding boxes.

As the first step toward the region-wise attention, we refine the erroneous depth map by using the clean RGB feature. A single-channel *pixel-wise* weighting map is derived from the RGB features through a $1 \times 1$ convolution, $I = \sigma(\text{Conv}_{1\times1}(F^{rgb})) \in \mathbb{R}^{h \times w \times 1}$, where $\sigma$ is a ReLU activation that ensures all the refined depth values to be positive. Accordingly, the refined depth map $\hat{D}$ is then obtained by multiplying the pixel-wise weighting map, which is upsampled to have the same size as the depth map, $\hat{D} = \text{Down}(D \otimes \text{Up}(I)) \in \mathbb{R}^{h \times w \times 1}$, where $\otimes$ is the element-wise multiplication, and Down and Up denote downsampling and upsampling, respectively. Next, the refined depth map $\hat{D}$ is used to form a region-wise attention. Specifically, $\hat{D}$ is divided into $m$ binary mask maps $\hat{D}_i$, each of which is set to 1 for its corresponding depth range $(\frac{i-1}{m}, \frac{i}{m})$. The region-wise attention is defined by

[2] We remove the classification layer in ResNet-50 as a backbone. Any pretrained backbone can also be used for the Siamese structure.

$$\mathrm{Att}_{\mathrm{R}}(F) = \overbrace{F}^{\text{feature map}} \otimes \overbrace{\sum_{i=1}^{m} \hat{D}_i \Big( \sum_{j=1}^{hw} \big(\hat{D}_{i,j} \times \mathcal{S}(F^{rgb})_j\big) \Big/ \sum_{j=1}^{hw} \hat{D}_{i,j} \Big)}^{\text{region-wise weighting}}, \tag{1}$$

where $\mathcal{S}(F^{rgb})$ is the global saliency map [46] of the RGB feature. The pixels for an object not only share similar depth values (i.e., a binary mask) but also exhibit large saliency values (i.e., a global saliency), thus multiplying them helps finding salient object regions in the depth map. Last, the region-wise attention is applied to both the RGB and depth features along with the standard channel attention [40], thereby obtaining two refined features,

$$\hat{F}^{rgb} = [\mathrm{Att}_{\mathrm{C}}(\mathrm{Att}_{\mathrm{R}}(F^{rgb})), F^{rgb}] \text{ and } \hat{F}^{depth} = [\mathrm{Att}_{\mathrm{C}}(\mathrm{Att}_{\mathrm{R}}(F^{depth})), F^{depth}], \tag{2}$$

where $\mathrm{Att}_{\mathrm{C}}$ and $[\cdot]$ are channel-wise attention and feature concatenation along the channel. We present the qualitative analysis of using the feature refiner in Section 2.1 of the supplementary material.

## 3.3 Multimodal Transformer [3]

**Encoder: Modality-Wise Self-Attention** The role of the encoder is to perform global scene reasoning for each modality, finding the areas of the feature map that can contribute to object detection. Thus, MEDUSA allocates a dedicated self-attention block for each modality, which we call *modality-wise self-attention*.

Two $1 \times 1$ convolutions reduce the channel dimension of each refined feature map from $2c$ to $d$ ($d < 2c$), generating two compact feature maps. Then, we flatten the spatial contexts by reshaping into a one-dimensional sequence of length $hw$ for the input to the transformer,

$$Z^{rgb} = (\mathbf{z}_1^{rgb}, \mathbf{z}_2^{rgb}, \ldots, \mathbf{z}_{hw}^{rgb}) \text{ and } Z^{depth} = (\mathbf{z}_1^{depth}, \mathbf{z}_2^{depth}, \ldots, \mathbf{z}_{hw}^{depth}) \text{ where } \forall_i\, \mathbf{z}_i \in \mathbb{R}^d. \tag{3}$$

These two embedding sequences are fed to *two* self-attention streams for their respective modalities, sharing the same self-attention module with a point-wise feed-forward network (FFN). The shared self-attention module recognizes all pairwise point interactions, being formulated by the self-attention map SA,

$$\mathrm{SA}(Z) = \mathrm{Softmax}\big((ZW_Q)(ZW_K)^\top / \sqrt{d}\big) \in \mathbb{R}^{hw \times hw} \text{ where } Z \in \{Z^{rgb}, Z^{depth}\}, \tag{4}$$

and then an embedding sequence $Z$ is encoded by $Z' = \mathrm{SA}(Z)(ZW_V)$, where $W_Q, W_K$, and $W_V$ are the query, key, and value projection matrices of the self-attention module.

Each encoder layer follows the standard form of the multi-head attention mechanism. Before entering the self-attention module, the sinusoidal-based spatial positional encoding [2, 48] is added to the input of each attention layer to supplement the permutation-invariant in the transformer. All encodings from the parallel heads are concatenated and fed to the point-wise FFN together with residual connections and layer normalization, which are commonly observed in the transformer [35]. Next, the two modality-wise representations $H^{rgb}$ and $H^{depth}$ are obtained and stacked along the sequence dimension, $H^{rgbd} = [H^{rgb}, H^{depth}]^\top \in \mathbb{R}^{(hw_{rgb} + hw_{depth}) \times d}$. Note that we do not perform RGB-D fusion at the encoder level because the fusion strategy should be differentiated for each object.

**Decoder: Object-Level RGB-D Fusion** The role of the decoder is to infer the best RGB-D fusion policy and then produce the final object embeddings. In contrast to the previous

[3] We describe a multimodal transformer architecture with a single-layer encoder and decoder just for ease of exposition. It can be extended to the $L$-layer structure by stacking them sequentially just like the standard transformer. Please refer to Section 4 of the supplementary material for readers unfamiliar with the transformer.

work [22, 31, 42] in which RGB and depth features are fused in a batch by adding or concatenating the entire feature maps, to achieve the most appropriate RGB-D fusion, the multimodal attention block in the decoder differentiates the fusion strategy for each (predicted) object provided by the object query (see the analysis of the object-level fusion in Section 4.3).

Similar to DETR [2], the decoder processes $n$ objects in parallel where $n$ is the number of objects to detect per image. The input of the decoder is called *object queries*, which are the learned encodings to help the decoder produce diverse results (i.e., bounding boxes and labels) by adding themselves to the query input of each attention layer. The $n$ object queries are first converted to an intermediate representation $H^{obj} \in \mathbb{R}^{n \times d}$ through the multi-head self-attention module. Next, for the *object-level* RGB-D fusion, the *multimodal* attention module receives the intermediate representation $H^{obj}$ for $n$ objects as its query and the stacked sequence $H^{rgbd}$ as its key and value. Thus, for individual query (object), the RGB-D fusion strategy is derived in the form of a multimodal attention map MA, which is the softmax of dot products between the query and key,

$$\mathrm{MA}(H^{obj}, H^{rgbd}) = \mathrm{Softmax}\big((H^{obj}W'_Q)(H^{rgbd}W'_K)^\top/\sqrt{d}\big) \in \mathbb{R}^{n \times (hw_{rgb}+hw_{depth})}, \tag{5}$$

where $W'_Q, W'_K$, and $W'_V$ be the query, key, and value projection matrices of the multimodal attention block. Therefore, the RGB-D object embedding for $n$ objects are computed by $O^{obj} = \mathrm{MA}(H^{obj}, H^{rgbd})(H^{rgbd}W'_V) \in \mathbb{R}^{n \times d}$. The point-wise FFN and layer normalization layer are followed for the final object embedding, which is fed to a 3-layer FFN for bounding box regression and a linear projection layer for classification.

# 4 Experiments

We show that MEDUSA achieves much higher detection performance on three benchmark data than DETR and its three extensions for RGB-D fusion. Then, we conduct a detailed ablation study of our proposed framework. Finally, we provide insights into why the object-level fusion plays an important role in detection and ultimately improves the performance.

**Datasets** Three benchmark datasets are used for evaluation: Pascal VOC, Microsoft COCO, and SUN-RGBD – an instance of data contains 2.35, 6.53, and 5.66 objects on average, respectively. We apply the state-of-the-art monocular depth estimator, MiDaS [27], for VOC and COCO because ground-truth depth maps are not available in both datasets. Please see Section 3.1 of the supplementary material for the details.

**Algorithms** The existing RGB-D object detection methods in Table 1 are, in fact, not directly comparable with MEDUSA: (1) they require stereo images (HHA format), which are absent in VOC and COCO, and (2) they employ different types of detectors, resulting in an unfair comparison. Thus, we injected their underlying RGB-D fusion philosophy into DETR [2]. Following the recent work [22], only the layers compatible with the RGB pre-trained model were fine-tuned for these DETR extensions.

- Early Fusion: The depth map is treated as the fourth channel of the RGB input. The backbone was fine-tuned from the pretrained weight except the first four-channel input layer.
- Late Fusion V1: Maintaining a two-stream backbone for each modality. The RGB backbone was fine-tuned, while the depth backbone was trained from scratch.
- Late Fusion V2: This variant is similar to Late Fusion V1, but the depth backbone was fine-tuned in the same way as Early Fusion.

Overall, the baseline, DETR (RGB), and its three extensions for RGB-D fusion were compared against the MEDUSA approach. For SUN-RGBD data where ground-truth depth is available, MEDUSA is further compared to the three existing RGB-D detection methods.

| Method | Aero | Bike | Bird | Boat | Bottle | Bus | Car | Cat | Chair | Cow |
|---|---|---|---|---|---|---|---|---|---|---|
| DETR (RGB) | 79.7 | **89.5** | 80.0 | 77.2 | 54.0 | 90.6 | 89.4 | 90.3 | 68.1 | 88.8 |
| Early Fusion | 79.1 | 88.6 | **80.1** | 76.7 | 52.5 | **91.2** | 89.1 | 90.2 | 76.1 | 81.1 |
| Late Fusion V1 | 79.7 | 80.1 | 75.1 | 66.9 | 35.2 | 72.9 | 78.8 | **91.6** | 51.6 | 78.2 |
| Late Fusion V2 | 78.6 | 87.6 | 78.9 | 75.5 | 57.1 | 88.9 | 88.4 | 89.6 | 65.9 | 88.4 |
| **MEDUSA** | **80.2** (+0.5) | 89.2 (-0.3) | 79.9 (-0.1) | **77.4** (+0.2) | **60.2** (+6.2) | 90.2 (-0.4) | **89.5** (+0.1) | 90.6 (+0.3) | **76.3** (+8.2) | **89.3** (+0.5) |

(a) First ten classes among 20 classes.

| Method | Table | Dog | Horse | Mbike | Person | Plant | Sheep | Sofa | Train | TV | $AP_{50}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| DETR (RGB) | 80.7 | 90.1 | 90.4 | 88.4 | 79.0 | 61.1 | 87.0 | 79.1 | 90.3 | 80.7 | 81.7 |
| Early Fusion | **81.7** | 89.6 | 89.7 | 79.5 | 79.0 | 63.9 | 87.4 | 88.5 | 89.8 | 80.9 | 81.7 |
| Late Fusion V1 | 72.4 | **90.3** | 81.1 | 78.8 | 73.2 | 39.3 | 67.6 | 80.9 | **90.5** | 75.8 | 73.0 |
| Late Fusion V2 | 79.5 | 89.8 | 89.6 | 87.0 | 77.8 | 60.9 | 87.3 | **88.7** | 89.5 | 88.3 | 81.9 |
| **MEDUSA** | 79.8 (-0.9) | 90.2 (+0.1) | 90.3 (-0.1) | **88.7** (+0.3) | **79.2** (+0.2) | **65.2** (+4.1) | **87.7** (+0.7) | 79.5 (+0.4) | 90.3 (0.0) | **88.7** (+8.0) | **83.1** (+1.4) |

(b) Last ten classes among 20 classes along with the $AP_{50}$ on the entire classes.

Table 2: **Results for VOC data.** The last column indicates the mAP at IoU 0.5 ($AP_{50}$) on the entire classes. The highest value for each class is marked in bold, and the value inside the parentheses in MEDUSA denotes the difference from DETR (RGB).

| Resolution | Method | Epochs | $AP_{50:95}$ | $AP_{50}$ | $AP_{75}$ | $AP_S$ | $AP_M$ | $AP_L$ |
|---|---|---|---|---|---|---|---|---|
| $300 \times 500$ | DETR (RGB) | 150 | 27.4 | 45.8 | 27.5 | 6.2 | 27.2 | 49.2 |
| | Early Fusion | 150 | 26.4 | 44.8 | 26.1 | 6.2 | 25.4 | 47.3 |
| | Late Fusion V1 | 150 | 26.6 | 44.4 | 27.0 | 6.3 | 25.8 | 48.0 |
| | Late Fusion V2 | 150 | 28.1 | 46.5 | 28.0 | 6.5 | 28.1 | 50.3 |
| | **MEDUSA** | 150 | **28.9** (+1.5) | **47.1** (+1.3) | **29.2** (+1.7) | **7.8** (+1.6) | **28.7** (+1.5) | **51.3** (+2.1) |
| $420 \times 700$ | DETR (RGB) | 150 | 32.5 | 52.7 | 33.4 | 10.1 | 34.2 | 54.4 |
| | **MEDUSA** | 150 | **33.6** (+1.1) | **53.5** (+0.8) | **34.3** (+0.9) | **11.4** (+1.3) | **35.5** (+1.3) | **55.9** (+1.5) |
| $800 \times 1333$ | DETR (RGB) | 150 | 38.0 | 58.9 | 39.6 | 16.7 | 40.8 | 58.2 |
| | **MEDUSA** | 150 | **40.0** (+2.0) | **60.8** (+1.9) | **41.6** (+2.0) | **18.1** (+1.4) | **43.1** (+2.3) | **60.0** (+1.8) |
| $800 \times 1333$ | DETR (RGB) | 500 | 42.0 | 62.3 | 44.2 | 20.5 | 45.8 | 61.1 |
| | **MEDUSA** | 500 | **42.8** (+0.8) | **63.3** (+1.0) | **44.7** (+0.5) | **22.2** (+1.7) | **48.0** (+2.2) | **65.1** (+4.0) |

Table 3: **Results for COCO data.** [4] The mAP over multiple thresholds are summarized for three image resolutions.

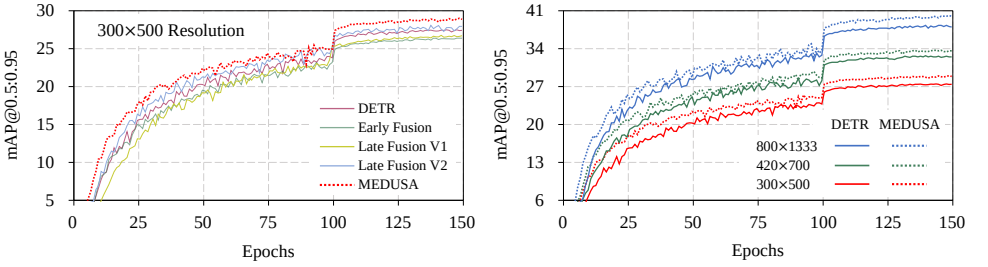
**Implementation Details**

All the algorithms were implemented using PyTorch and executed using eight NVIDIA V100 GPUs. They were trained for 150 epochs using AdamW [18] with a weight decay of $10^{-4}$ and a dropout of 0.1. The ResNet-50 pretrained on ImageNet was used as the backbone, and all transformer weights were initialized with Xavier initialization. We used an initial learning rate of $10^{-4}$ decayed by 10 at the 100-th epoch except the ResNet-50 backbone. An initial learning rate of $10^{-5}$ was used for fine-tuning.

For VOC and SUN-RGBD, the image resolution was $600 \times 1000$, and the batch size was set to 16; for COCO, the image resolution was resized to $300 \times 500$, $420 \times 700$, $800 \times 1333$ to demonstrate consistent improvements over varying resolutions, and the batch size was set to 32. In addition, random crops and horizontal flips were applied for data augmentation. More details can be found in Section 3.2 of the supplementary material. The source code and trained models are publicly available at https://github.com/songhwanjun/MEDUSA.

## 4.1 Performance Evaluation

**Pascal VOC Dataset** Table 2 shows the experimental results on the Pascal VOC dataset. Although the inferred depth is fairly noisy, the performance is improved significantly by MEDUSA compared with the baseline; the mAP at IoU 0.5 ($AP_{50}$) of MEDUSA is 83.1%,




(a) Comparison with four methods. (b) Comparison with varying resolutions.

Figure 3: **Convergence curve.** The mAP over training epochs for COCO val set are plotted.

which is higher than that of DETR (RGB) by 1.4%. The direct use of the inferred depth with the early or late fusion is not that effective; the $AP_{50}$ of Early Fusion or Late Fusion V2 is close to that of the baseline, while that of Late Fusion V1 rather drops by 8.7%.

MEDUSA achieves the best results in 14 classes out of the 20 classes. In particular, the performance gains in the "bottle," "chair," "plant," and "TV" classes are significant; the $AP_{50}$ of those classes are respectively increased by 6.2%, 8.2%, 4.1%, and 8.0% compared with DETR (RGB). Meanwhile, it is not effective to achieve significant gains with the early and late fusion approaches for the following reasons: (i) they do not handle the noisy depth caused by the estimation process; (ii) the RGB and depth features are simply concatenated, but this implicit fusion makes the model biased toward only RGB information as the RGB information mostly dominates the final prediction [6]; and (iii) the pretrained model is not fully used because of the difference in framework and modality, and considerable degradation of Late Fusion V1 is attributed to the mismatch between the two backbone streams. MEDUSA address these issues through its three components—Siamese structure, feature refiner, and multimodal transformer for object-level RGB-D fusion.

**Microsoft COCO Dataset** Table 3 shows the mAP results with different thresholds on the Microsoft COCO dataset. Further, to clearly verify the performance improvement, we provide the mAP convergence curve of MEDUSA along with compared methods with varying resolutions in Figure 3. When all the five methods are evaluated on images of $300 \times 500$ pixels, the general trend is similar to that for VOC. MEDUSA achieves the highest mAP over multiple thresholds and object scales, which are 1.3%–2.1% higher than DETR (RGB). In contrast, the performance of Early Fusion and Late Fusion V1 drops by 0.2%–1.7% and 0.1%–1.6%. Meanwhile, Late Fusion V2 achieves 0.1%–1.0% higher mAP than DETR, which confirms that the weights pretrained from RGB images are useful even for the depth backbone. Even for the $420 \times 700$ and $800 \times 1333$ resolutions, MEDUSA still achieves the consistent improvement of 0.8%–1.5% and 0.5%–4.0% compared with DETR (RGB), respectively. In terms of the mAP convergence, MEDUSA exhibits consistently higher values over *all* training epochs compared with other methods, as shown in Figure 3(a), and compared with DETR at varying resolutions, as in Figure 3(b). Therefore, the use of inferred depth maps with MEDUSA results in a more effective object detector.

**SUN-RGBD Dataset** We evaluate the improvements in SUN-RGBD data when using inferred and ground-truth depth in Table 4(a). Except for Early Fusion, all methods obtain performance improvement over the DETR (RGB) with the inferred depth. However, these schemes achieve significantly higher mAP with the ground-truth depth. In particular,

[4]The performance of all methods improves significantly with the increase in the image resolution. We also experiment with the resolution of $800 \times 1333$ pixels used in the DETR paper using two different learning schedules.

| Type | N/A | Estimated Depth | | | | Ground-Truth Depth | | | |
|---|---|---|---|---|---|---|---|---|---|
| Method | DETR (RGB) | Early Fus. | Late Fus. V1 | Late Fus. V2 | **MEDUSA** | Early Fus. | Late Fus. V1 | Late Fus. V2 | **MEDUSA** |
| $AP_{50}$ | 56.9 | 54.7 | 57.5 | 57.9 | **59.7** | 55.3 | 58.1 | 58.5 | **60.9** |

(a) Performance difference when using estimated and ground-truth depth maps.

| Backbone | AlexNet | | | | ResNet-50 |
|---|---|---|---|---|---|
| Method | RGB-D RCNN [10] | RGB-D RPN [10] | RGB-D F-RCNN [10] | **MEDUSA** | **MEDUSA** |
| $AP_{50}$ | 32.9 | 51.8 | 52.9 | **54.3** | **60.9** |

(b) Performance comparison with three existing methods for RGB-D object detection.

Table 4: **Results for SUN-RGBD data.** The results for the existing methods are borrowed from [10, 41] since there is no source code available. The highest value is marked in bold.

| | Feature Extractor | | Feature Refiner | | | RGB-D Fusion | | Measure | | |
|---|---|---|---|---|---|---|---|---|---|---|
| # | Two-stream | Siamese | Region Att. | Pixel Rew. | Channel Att. | Scene-Level | Object-Level | $AP_{50}$ | Params. | Hour |
| (1) | ✓ | | | | | ✓ | | 77.3 | 65.2M | 25.3 |
| (2) | | ✓ | | | | ✓ | | 77.0 | 41.8M | 20.9 |
| (3) | | ✓ | ✓ | | | ✓ | | 77.1 | 42.8M | 21.3 |
| (4) | | ✓ | ✓ | ✓ | | ✓ | | 77.3 | 42.9M | 22.7 |
| (5) | | ✓ | ✓ | ✓ | ✓ | ✓ | | 77.6 | 43.9M | 22.8 |
| (6) | | ✓ | ✓ | ✓ | ✓ | | ✓ | 78.5 | 43.9M | 23.5 |

Table 5: **Detailed ablation study using VOC data with 300×400 resolution.**

MEDUSA achieves the largest performance improvement in both cases, and outperforms other DETR variants by 1.8%–5.6%. Our method can handle a variety of noise even in the ground-truth depth map, including missing estimates (holes) and noisy boundaries. We expect more performance gains can be achieved by using more accurate depth estimators such as DPT [28].

Table 4(b) shows evaluation results against three RGB-D detection methods when using the ground-truth depth. With the same configuration using the AlexNet backbone, MEDUSA outperforms these schemes by 1.4%–21.4%. When using the ResNet-50, the performance of MEDUSA reaches 60.9% owing to a more effective feature representation model.

## 4.2 Ablation Study

**Extension with Deformable DETR** The DETR detector in MEDUSA can be easily replaced with other state-of-the-art detectors, e.g., Deformable DETR [48]. With the same configuration for VOC dataset in Table 2, the $AP_{50}$ of our method with Deformable DETR is 86.7%, which is 1.0% and 3.6% higher than those of RGB-only Deformable DETR and MEDUSA with DETR, respectively. The depth estimation module plays an important role to help increase the performance of detectors, even when using the state-of-the-art model.

**Contribution of Each Component** Table 5 shows the effectiveness of all design choices for MEDUSA on the VOC dataset using images of 300×400 pixels. The object-level fusion via the multimodal transformers (6) obtains the biggest gain of 0.9% over the scene-level fusion (feature concatenation). The model using region-wise attention with the pixel reweighting (4) performs better than that using the region-wise attention alone (3). The channel-wise attention (5) further improves the performance. The two-stream extractor (1) performs slightly better than the Siamese extractor (2), but with heavy computational load.

**Analysis of Model Parameters** The parameter increase in MEDUSA is much smaller than that in representative previous work [22, 42]; specifically, the late fusion approach adds

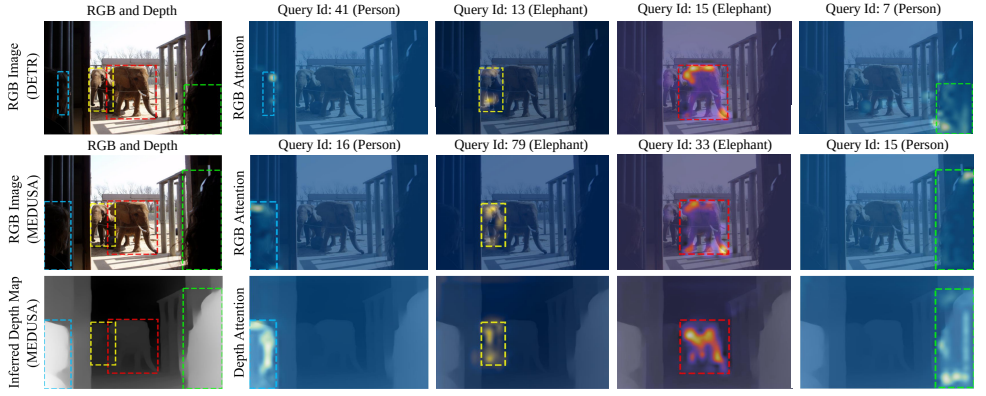



Figure 4: **Visualization of the RGB-D fusion policy for detected objects.** The first row shows all the attention weights for RGB in DETR(RGB), and the second and third rows show those for RGB and Depth derived from the "multimodal attention block" in MEDUSA. The attention weights and predicted bounding boxes are coded in different colors for each object.

23.9M parameters to the 41.3M parameters used for the original DETR, while MEDUSA adds only 2.6M parameters because of its shared structure. The performance gain mainly comes from our three components rather than a larger number of parameter usage.

### 4.3 Analysis of Object-Level Fusion

We analyze the object-level RGB-D fusion (the attention weights) of the multimodal transformer for RGB-D fusion. MEDUSA determines a different fusion policy for each object (query) such that it gives different weights to RGB and depth features for better detection. Figure 4 shows the attention map and predicted bounding box of MEDUSA for the objects that DETR (RGB) fails to detect. The two people enclosed by cyan and green boxes are not easily recognizable from the RGB images (i.e., the first and second rows), owing to the similar color with the background and overlap with another object. Thus, MEDUSA gives dominant weights to depth features to help recognizing them via RGB-D fusion. However, it turns out that depth attention could leak out of the bounding box unlike RGB (e.g., the query 16). That is, only using the depth input is not sufficient for box regression. Thus, the depth input should be used in conjunction with the RGB input for better object detection.

Moreover, it is interesting that MEDUSA pays attention to the overall appearance of an object in depth view, whereas to some discriminative parts (e.g., head and leg) in RGB view for RGB-D fusion. More examples of the attention maps are presented in Section 2.2 of the supplementary material.

## 5 Conclusion

In this paper, we have proposed a generic object detection framework for leveraging the estimated depth. The Siamese network and feature refiner are able to generate high-quality RGB and depth features; then, the multimodal transformer determines the complementary fusion strategy between RGB and depth features via the cross-attention mechanism. This model can exploit depth information even for regular RGB images without requiring the ground-truth depth. Using the three benchmark datasets, the proposed approach is shown to outperform the DETR extension and three existing methods. In addition, detailed ablation studies are presented together with an insight into the object-level fusion.

# References


[1] Yuanzhouhan Cao, Chunhua Shen, and Heng Tao Shen. Exploiting depth from single monocular images for object detection and semantic segmentation. *IEEE Transactions on Image Processing*, 26(2):836–846, 2016.

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, pages 213–229, 2020.

[3] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

[4] Deng-Ping Fan, Zheng Lin, Zhao Zhang, Menglong Zhu, and Ming-Ming Cheng. Rethinking RGB-D salient object detection: Models, data sets, and large-scale benchmarks. *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[5] Deng-Ping Fan, Yingjie Zhai, Ali Borji, Jufeng Yang, and Ling Shao. BBS-Net: RGB-D salient object detection with a bifurcated backbone strategy network. In *ECCV*, pages 275–292, 2020.

[6] Keren Fu, Deng-Ping Fan, Ge-Peng Ji, and Qijun Zhao. JL-DCF: Joint learning and densely-cooperative fusion framework for RGB-D salient object detection. In *CVPR*, pages 3052–3062, 2020.

[7] Saurabh Gupta, Ross Girshick, Pablo Arbeláez, and Jitendra Malik. Learning rich features from RGB-D images for object detection and segmentation. In *ECCV*, pages 345–360, 2014.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[9] Saihui Hou, Zilei Wang, and Feng Wu. Deeply exploit depth information for object detection. In *CVPRW*, pages 19–27, 2016.

[10] Saihui Hou, Zilei Wang, and Feng Wu. Object detection via deeply exploiting depth information. *Neurocomputing*, 286:58–66, 2018.

[11] Jianbo Jiao, Yunchao Wei, Zequn Jie, Honghui Shi, Rynson WH Lau, and Thomas S Huang. Geometry-aware distillation for indoor semantic segmentation. In *CVPR*, pages 2869–2878, 2019.

[12] Jae-Han Lee and Chang-Su Kim. Monocular depth estimation using relative depth maps. In *CVPR*, pages 9729–9738, 2019.

[13] Gongyang Li, Zhi Liu, and Haibin Ling. ICNet: Information conversion network for RGB-D based salient object detection. *IEEE Transactions on Image Processing*, 29: 4873–4884, 2020.

[14] Yabei Li, Zhang Zhang, Yanhua Cheng, Liang Wang, and Tieniu Tan. MAPNet: Multi-modal attentive pooling network for RGB-D indoor scene classification. *Pattern Recognition*, 90:436–449, 2019.

[15] Zhengqi Li and Noah Snavely. MegaDepth: Learning single-view depth prediction from internet photos. In *CVPR*, pages 2041–2050, 2018.

[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, pages 740–755, 2014.

[17] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. SSD: Single shot multibox detector. In *ECCV*, pages 21–37, 2016.

[18] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

[19] Ao Luo, Xin Li, Fan Yang, Zhicheng Jiao, Hong Cheng, and Siwei Lyu. Cascade graph neural networks for RGB-D salient object detection. In *ECCV*, pages 346–364, 2020.

[20] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *CVPR*, pages 3061–3070, 2015.

[21] Irina Mocanu and Cosmin Clapon. Multimodal convolutional neural network for object detection using RGB-D images. In *IEEE Transactions on Signal Processing*, pages 1–5, 2018.

[22] Tanguy Ophoff, Kristof Van Beeck, and Toon Goedemé. Exploring RGB+ depth fusion for real-time object detection. *Sensors*, 19(4):866, 2019.

[23] Seong-Jin Park, Ki-Sang Hong, and Seungyong Lee. RDFNet: RGB-D multi-level residual feature fusion for indoor semantic segmentation. In *ICCV*, pages 4980–4989, 2017.

[24] Andrea Pilzer, Stephane Lathuiliere, Nicu Sebe, and Elisa Ricci. Refine and distill: Exploiting cycle-inconsistency and knowledge distillation for unsupervised monocular depth estimation. In *CVPR*, pages 9768–9777, 2019.

[25] Matteo Poggi, Filippo Aleotti, Fabio Tosi, and Stefano Mattoccia. On the uncertainty of self-supervised monocular depth estimation. In *CVPR*, pages 3227–3237, 2020.

[26] Wasifur Rahman, Md Kamrul Hasan, Sangwu Lee, AmirAli Bagher Zadeh, Chengfeng Mao, Louis-Philippe Morency, and Ehsan Hoque. Integrating multimodal information in large pretrained transformers. In *ACL*, pages 2359–2369, 2020.

[27] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[28] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. *arXiv preprint arXiv:2103.13413*, 2021.

[29] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, pages 779–788, 2016.

[30] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NeurIPS*, pages 91–99, 2015.

[31] Max Schwarz, Anton Milan, Arul Selvam Periyasamy, and Sven Behnke. RGB-D object detection and semantic segmentation for autonomous manipulation in clutter. *International Journal of Robotics Research*, 37(4-5):437–451, 2018.

[32] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from RGBD images. In *ECCV*, pages 746–760, 2012.

[33] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun RGB-D: A RGB-D scene understanding benchmark suite. In *CVPR*, pages 567–576, 2015.

[34] Yao-Hung Hubert Tsai, Shaojie Bai, Paul Pu Liang, J Zico Kolter, Louis-Philippe Morency, and Ruslan Salakhutdinov. Multimodal transformer for unaligned multimodal language sequences. In *ACL*, volume 2019, page 6558, 2019.

[35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pages 5998–6008, 2017.

[36] Weiyue Wang and Ulrich Neumann. Depth-aware CNN for RGB-D segmentation. In *ECCV*, pages 135–150, 2018.

[37] Xinlong Wang, Wei Yin, Tao Kong, Yuning Jiang, Lei Li, and Chunhua Shen. Task-aware monocular depth estimation for 3d object detection. In *AAAI*, 2020.

[38] Yan Wang, Wei-Lun Chao, Divyansh Garg, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In *CVPR*, 2019.

[39] Yisen Wang, Weiyang Liu, Xingjun Ma, James Bailey, Hongyuan Zha, Le Song, and Shu-Tao Xia. Iterative learning with open-set noisy labels. In *CVPR*, pages 8688–8696, 2018.

[40] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. CBAM: Convolutional block attention module. In *ECCV*, pages 3–19, 2018.

[41] Xiangyang Xu, Yuncheng Li, Gangshan Wu, and Jiebo Luo. Multi-modal deep feature learning for RGB-D object detection. *Pattern Recognition*, 72:300–313, 2017.

[42] Jiachen Yang, Chenguang Wang, Huihui Wang, and Qiang Li. A RGB-D based real-time multiple object detection and ranging system for autonomous driving. *IEEE Sensors Journal*, 2020.

[43] Shaowei Yao and Xiaojun Wan. Multimodal transformer for multimodal machine translation. In *ACL*, pages 4346–4350, 2020.

[44] Yuan Yuan, Zhitong Xiong, and Qi Wang. ACM: Adaptive cross-modal graph convolutional neural networks for RGB-D scene recognition. In *AAAI*, volume 33, pages 9176–9184, 2019.

[45] Jing Zhang, Deng-Ping Fan, Yuchao Dai, Saeed Anwar, Fatemeh Sadat Saleh, Tong Zhang, and Nick Barnes. UC-Net: Uncertainty inspired RGB-D saliency detection via conditional variational autoencoders. In *CVPR*, pages 8582–8591, 2020.

[46] Miao Zhang, Weisong Ren, Yongri Piao, Zhengkun Rong, and Huchuan Lu. Select, supplement and focus for RGB-D saliency detection. In *CVPR*, pages 3472–3481, 2020.

[47] Wujie Zhou, Jianzhong Yuan, Jingsheng Lei, and Ting Luo. TSNet: Three-stream self-attention network for RGB-D indoor semantic segmentation. *IEEE Intelligent Systems*, 2020.

[48] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: Deformable transformers for end-to-end object detection. In *ICLR*, 2021.