

# MatchDet: A Collaborative Framework for Image Matching and Object Detection

Jinxiang Lai <sup>\*1</sup>, Wenlong Wu <sup>\*1</sup>, Bin-Bin Gao <sup>1</sup>, Jun Liu <sup>1</sup>, Jiawei Zhan <sup>1</sup>, Congchong Nie <sup>1</sup>, Yi Zeng <sup>1</sup>, Chengjie Wang <sup>† 1,2</sup>

<sup>1</sup> Tencent Youtu Lab, China

<sup>2</sup> Shanghai Jiao Tong University, China

layjins1994@gmail.com, ezrealwu@tencent.com, {csgaobb, junsenselee}@gmail.com, {gavynzhan, nickccnie, sylviazeng}@tencent.com, jasoncjwang@tencent.com

## Abstract

Image matching and object detection are two fundamental and challenging tasks, while many related applications consider them two individual tasks (i.e. task-individual). In this paper, a collaborative framework called MatchDet (i.e. task-collaborative) is proposed for image matching and object detection to obtain mutual improvements. To achieve the collaborative learning of the two tasks, we propose three novel modules, including a Weighted Spatial Attention Module (WSAM) for Detector, and Weighted Attention Module (WAM) and Box Filter for Matcher. Specifically, the WSAM highlights the foreground regions of target image to benefit the subsequent detector, the WAM enhances the connection between the foreground regions of pair images to ensure high-quality matches, and Box Filter mitigates the impact of false matches. We evaluate the approaches on a new benchmark with two datasets called Warp-COCO and miniScan-Net. Experimental results show our approaches are effective and achieve competitive improvements.

## Introduction

Image matching (Shrivastava et al. 2011) and object detection (Liu et al. 2020) are two fundamental and challenging tasks in computer vision. Image matching finds pixel-wise correspondences between image pairs, and object detection seeks to locate and classify object instances in images. With the combination of them, there are numerous important applications, including robot vision, autonomous driving, and industrial defect inspection. In robot vision and autonomous driving, it usually uses image matching technique to perform Simultaneous Localization And Mapping (SLAM) (Mur-Artal, Montiel, and Tardos 2015), and also needs to find the target category objects (e.g. pedestrian in autonomous driving) in images based on object detection technique. In industrial defect inspection, it applies image matching for registration (Shrivastava et al. 2011) to obtain Region-of-Interest (ROI), and then detects the target defects. The aforementioned applications consider image matching and object detection as two individual tasks (i.e. task-individual).

In this paper, a collaborative framework called MatchDet (i.e. task-collaborative) is proposed for image matching and

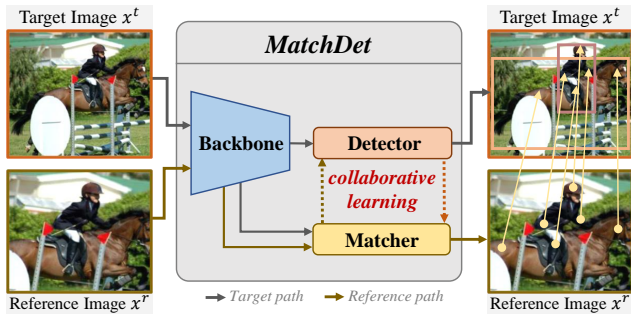
object detection to obtain mutual improvements. As illustrated in Fig.1(a), given input reference and target images, our MatchDet simultaneously outputs their homography relationship and object detection results of the target image, which is defined as a Match-and-Detection task. The proposed MatchDet framework consists of a shared backbone, and two Matcher and Detector task branches for image matching and object detection, which is co-trained end-to-end. Under task-collaborative MatchDet framework, the homography relationship estimated by Matcher and the bounding boxes predicted by Detector can be useful to each other.

As presented in Fig.1(b), the most relevant approach is the correlation based Tracker (Christoph, Axel, and Andrew 2017) for the object tracking task. The Tracker utilizes the Correlation module to explore the affinity between the target image and the reference image, then further applies the Aggregation module to refine the affinity for enhancing the target objects. There are two main differences between MatchDet and Tracker: (i) MatchDet integrates a Matcher branch, which is able to obtain a precise homography relationship. However, Tracker only adopts the Correlation module to implicitly explore the affinity without the supervision of matching objective function, which leads to an imprecise homography relationship. As illustrated in Fig.1(d), our MatchDet achieves a large improvement with 39.96% in image matching task. (ii) MatchDet achieves mutual performance improvements in the two tasks via the proposed collaborative learning module, while Tracker only utilizes correlation-aggregation learning to improve the performance of the object detection task.

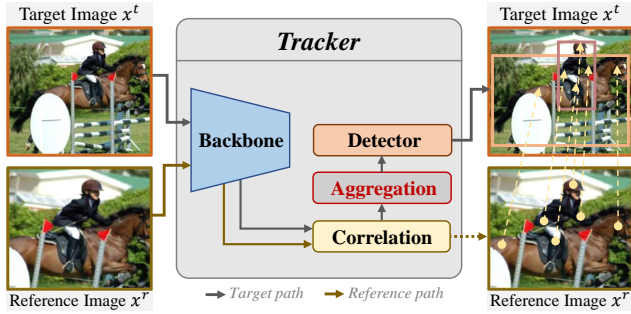
To achieve mutual performance improvements via the collaborative learning of the two tasks, we propose three novel modules, including a Weighted Spatial Attention Module (WSAM) for Detector, Weighted Attention Module (WAM) and Box Filter for Matcher. For Matcher branch, the proposed WAM produces more discriminative feature representations of image pairs, via learning global context to find the correspondences among surrounding regions, with the usage of Transformer structure (Sun et al. 2021). Benefiting from the Match-and-Detection task, it's achievable to obtain the potential foreground regions of images. Then the WAM can be more focus on interacting information among the foreground regions of pairs, which reduces background interference and ensures high-quality matches. Specifically,

<sup>\*</sup>These authors contributed equally.

<sup>†</sup>Corresponding Author



(a) MatchDet framework with collaborative learning



(b) Object Tracker framework with correlation-aggregation learning

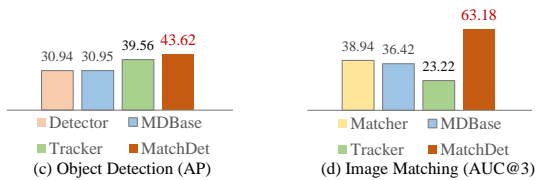


Figure 1: (a) Our MatchDet with collaborative learning for improving image matching and object detection. We introduce a baseline named MDBase network, which removes the collaborative learning module of MatchDet. (b) The object Tracker with correlation-aggregation learning. The dashed line represents that the Tracker has the potential ability to obtain pairwise correspondences, while there is no matching objective function to supervise it. (c) and (d) are the results on Warp-COCO dataset. (c) Our MatchDet obtains 4.06% improvement in object detection. (d) Our MatchDet achieves 24.24% higher performance in image matching.

the WAM first utilizes a Weights Generator to generate weighted maps based on the foreground regions, and then uses the weighted maps to enhance the affinity matrix between feature pairs as implementing in the Attention operation of Transformer. Furthermore, Box Filter reduces the impact of the potential low-quality matches, via strengthening the matching scores among the foreground regions of pairs, where the foregrounds are predicted by Detector.

For Detector branch, the proposed WSAM, a variant of WAM, highlights the foreground regions of target image, via the similar regions with instance feature of reference image and learnable semantic embedding. To achieve the above purpose of enhancing foregrounds, the WSAM adopts Weighted Spatial Attention instead of Weighted Attention

applied in WAM, and their essential difference is discussed in APPENDIX. Similar to WAM, the WSAM also pays more attention on feature interaction among foregrounds. The WSAM transforms the foreground regions of reference image with the homography estimated by Matcher, to generate the potential foreground regions for target image.

In general, our main contributions are:

- For the first time, we propose a collaborative framework called MatchDet for image matching and object detection to obtain mutual improvements. Besides, our MatchDet is a general framework, which can utilize different detectors such as FCOS, Faster-RCNN, and AdaMixer.

- To achieve collaborative learning of image matching and object detection, three novel modules are proposed, including a Weighted Spatial Attention Module (WSAM) which highlights the foreground regions of target image for Detector, and Weighted Attention Module (WAM) and Box Filter which obtains high-quality matches for Matcher.

- We evaluate the Match-and-Detection task on a new benchmark with two datasets called Warp-COCO and miniScanNet. This benchmark can be used to verify the performances of the algorithms on both image matching and object detection. Experimental results show our approaches are effective and achieve competitive improvements.

## Related Work

**Object Detection** The current object detection algorithms can be divided into anchor-based (Ren et al. 2015a; Redmon and Farhadi 2018; Lin et al. 2017b; Zhang et al. 2019, 2020b), anchor-free (Law and Deng 2018; Duan et al. 2019; Tian et al. 2019), and query-based (Carion et al. 2020; Gao et al. 2022; Zhang et al. 2023) methods. In this paper, we choose the classical anchor-free FCOS (Tian et al. 2019) as the basic detector due to its good performance and simplicity.

**Image Matching** Image matching finds pixel-wise correspondences between image pairs, with two main directions of Interest Point Detector-based (IPD-based) methods (Lowe 2004; Rublee et al. 2011; DeTone, Malisiewicz, and Rabinovich 2018) and IPD-free methods (Sun et al. 2021; Wang et al. 2022; Chen et al. 2022). In this paper, we choose the classical IPD-free LoFTR (Sun et al. 2021) as the basic image matcher due to its good performance and simplicity.

**Transformer Attention** The transformer Cross Attention in LoFTR (Sun et al. 2021) learns global context based on the full affinity matrix between feature pairs, but it may easily be distracted by background and causes the slow convergence of model. To alleviate this problem, we propose two novel transformer-based weighted attention and weighted spatial attention modules to strengthen the target object.

## Problem Definition

The Match-and-Detection task aims to obtain the homography relationship and object detection results of the input pair images. Formally, given the pair images  $x^t$  and  $x^r$ , the Match-and-Detection method predicts the homography matrix  $\hat{\mathcal{H}}$  and bounding boxes  $\{\hat{B}_i^t\}$  for  $x^t$ . We denote  $x^t$  and  $x^r$  as the target and reference images respectively, the

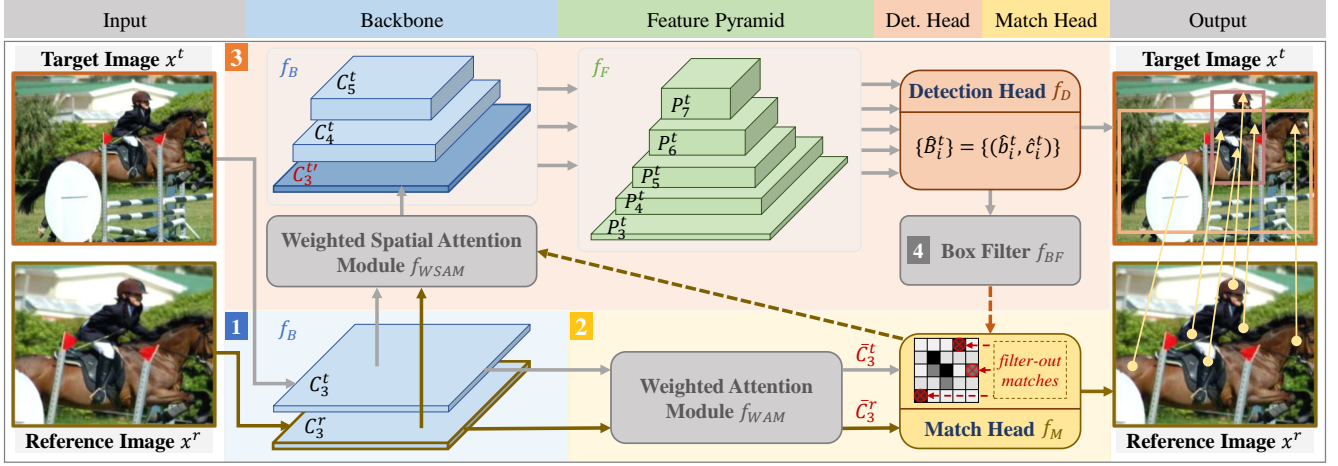


Figure 2: The network architecture of our MatchDet. There are four stages: ① Obtaining basic features  $\{C_3^t, C_3^r\}$  with a shared backbone. ② Matcher branch estimates the homography matrix with the enhanced features  $\{\bar{C}_3^t, \bar{C}_3^r\}$  produced by Weighted Attention Module. ③ Detector branch predicts the bounding boxes based on the highlighted features  $C_3^{t'}$  generated by Weighted Spatial Attention Module. ④ Box Filter refines the image matching results via filtering out the potential mismatches.

homography matrix  $\mathcal{H} \in \mathbb{R}^{3 \times 3}$  represents their geometric relationship as  $x^t = \mathcal{H}x^r$ , and their ground-truth bounding boxes are defined as  $\{B_i^t\}$  and  $\{B_i^r\}$  respectively. Here  $B_i = (b_i, c_i) \in \mathbb{R}^4 \times \{1, 2 \dots C\}$ ,  $b_i$  and  $c_i$  denote the bounding box and the category of the object respectively, and  $C$  is the classes number of dataset.

In training stage, all ground-truth labels are given. In inference stage, with different sources of bounding boxes of reference image, we introduce three different settings: (a) **GTBoxR** setting, gives Ground-Truth bounding Boxes  $\{B_i^r\}$  of Reference image in inference. Some of the corresponding applications are robotic Pick-and-Place (Zeng et al. 2018) and robot navigation for known scenarios, of which the reference image can be stored and labeled in advance. (b) **PreBoxR** setting, gives Prediction bounding Boxes  $\{\hat{B}_i^r\}$  of Reference image in inference. It is suitable for the video-based scenarios, which can reuse the prediction results of the previous frame image. (c) **NoBoxR** setting, gives No bounding Boxes of Reference image in inference. It is more general and challenging than the above settings.

## Methodology

We first propose a simple baseline method called MD-Base (Match-and-Detection Baseline) network  $f_{base}$ , which adopts multi-task paradigm with a shared backbone  $f_B$  and two task-heads (i.e. Match Head  $f_M$  and Detection Head  $f_D$ ) for matching and detection. Then, as illustrated in Fig.2, the MatchDet network  $f_{MD}$  is constructed upon MD-Base network, which additionally inserts three novel modules including Weighted Spatial Attention Module (WSAM)  $f_{WSAM}$ , Weighted Attention Module (WAM)  $f_{WAM}$  and Box Filter  $f_{BF}$ . The WSAM highlights the foreground regions of target image, the WAM enhances the connection between the foreground regions of pair images, and Box Filter mitigates the impact of false matches.

Specifically, let  $f_B^i$  be the  $i^{th}$  layer of backbone  $f_B$ ,  $f_F^i$  be the  $i^{th}$  layer of FPN (Feature Pyramid Network)  $f_F$ . Given input image pairs  $\{x^t, x^r\}$ , the backbone  $f_B$  generates corresponding target features  $\{C_3^t, C_4^t, C_5^t\}$  and reference features  $C_3^r$  respectively, and the FPN  $f_F$  further produces target features  $\{P_3^t, P_4^t, P_5^t, P_6^t, P_7^t\}$ . In particular, the  $i^{th}$  layer features  $C_i^t$  and  $C_i^r$  are produced by  $f_B^i$ , and the same goes for other features like  $P_i^t$  by  $f_F^i$ , where  $\{C_i^t, C_i^r, P_i^t\} \in \mathbb{R}^{c_i \times h_i \times w_i}$ . Then, WSAM and WAM generate the enhanced features  $C_3^{t'}$  and  $\{\bar{C}_3^t, \bar{C}_3^r\}$  for Detector and Matcher, respectively. Finally, the Match Head  $f_M$  and Detection Head  $f_D$  predict the homography matrix  $\hat{\mathcal{H}}$  and bounding boxes  $\{\hat{B}_i^t\}$  for  $x^t$ . In detail, the Detection Head  $f_D$  is the same as FCOS, and the Match Head  $f_M$  is similar to LoFTR which utilizes a dual-softmax matching layer.

## MDBase Network

The MDBase network consists of a shared backbone and two task-heads for matching and detection, i.e.  $f_{base} = [f_B, f_F, f_M, f_D]$ . The MDBase makes predictions as follows:

$$\begin{aligned} \{\hat{B}_i^t\}, \hat{\mathcal{H}} &= f_{base}(x^t, x^r) \\ \text{where, } \{\hat{B}_i^t\} &= f_D(f_F(f_B(x^t))), \\ \hat{\mathcal{H}} &= f_M(f_B(x^t, x^r)) \end{aligned} \quad (1)$$

Comparing to the current task-individual solutions, the MD-Base network has lower complexity by utilizing a shared backbone. In order to obtain mutual improvements of image matching and object detection, we next introduce a novel MatchDet network benefiting from the collaborative learning of the two tasks.

## MatchDet Network

As illustrated in Fig.2, on the basic of MDBase network, the proposed MatchDet integrates Weighted Spatial Attention

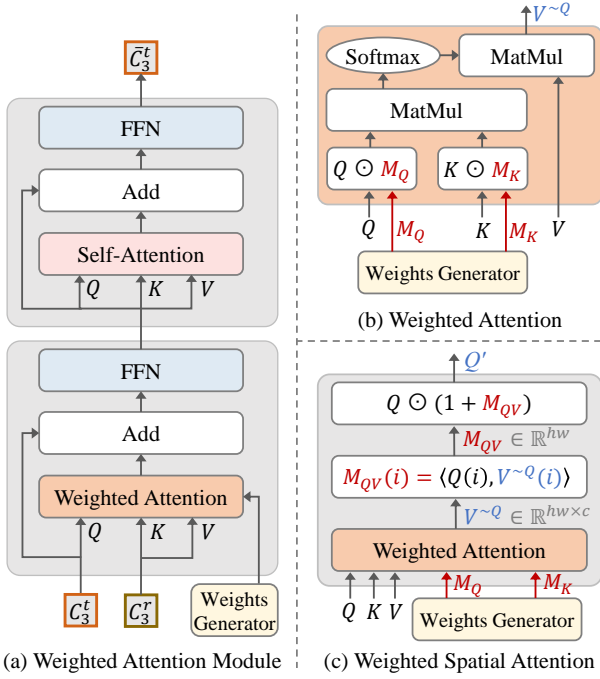


Figure 3: (a) The Weighted Attention Module (WAM) consists of a Weighted Attention block and a Self-Attention block, where  $\{Q, K, V\}$  are known as  $\{query, key, value\}$  and FFN denotes Feed-Forward Network in Transformer. (b) The Weighted Attention applied in WAM, where  $\odot$  is Broadcasting Element-wise Product. The variables dimensions are  $\{V^{\sim Q}, Q, K, V\} \in \mathbb{R}^{hw \times c}$  and  $\{M_Q, M_K\} \in \mathbb{R}^{hw}$ . (c) The Weighted Spatial Attention enhances the spatial response of  $Q$  by  $M_{QV} \in \mathbb{R}^{hw}$  to obtain  $Q' \in \mathbb{R}^{hw \times c}$ , where  $\langle \cdot \rangle$  calculates the cosine similarity. And replacing Weighted Attention of WAM with Weighted Spatial Attention derives the Weighted Spatial Attention Module (WSAM).



Figure 4: The visualizations of the generated Weighted Map.

Module  $f_{WSAM}$ , Weighted Attention Module  $f_{WAM}$  and Box Filter  $f_{BF}$  to achieve collaborative learning of image matching and object detection tasks. Formally, MatchDet consists of  $f_{MD} = [f_{base}, f_{WSAM}, f_{WAM}, f_{BF}]$ , and makes predictions as:

$$\{\hat{B}_i^t\}, \hat{H} = f_{MD}(x^t, x^r)$$

$$\text{where, } \{\hat{B}_i^t\} = f_D(f_F(f_{WSAM}(f_B(x^t, x^r))))), \quad (2)$$

$$\hat{H} = f_M(f_{BF}(f_{WAM}(f_B(x^t, x^r))))$$

In the following, we introduce the detailed structure of the proposed modules and loss function.

**Weighted Attention Module** The Weighted Attention Module  $f_{WAM}$  enhances the connection between the fore-

ground regions of pair images, which is expressed as:

$$\bar{C}_3^t, \bar{C}_3^r = f_{WAM}(C_3^t, C_3^r), f_{WAM}(C_3^r, C_3^t) \quad (3)$$

As shown in Fig.3(a), the WAM stacks a Weighted Attention block and a Self-Attention block. Different from the Cross Attention used in LoFTR (Sun et al. 2021) and Masked Attention applied in Mask2Former (Cheng et al. 2022), we propose a novel Weighted Attention as presented in Fig.3(b), which enhances  $\{Q, K\}$  with two different weighted maps  $\{M_Q, M_K\}$  produced by Weights Generator, respectively. The Weighted Attention operation is formulated as:

$$V^{\sim Q} = \text{WeightedAttention}((Q, K, V), M_Q, M_K)$$

$$= \sigma((Q \odot M_Q)(K \odot M_K)^T) V \quad (4)$$

where  $\sigma$  is softmax function. The weighted maps  $\{M_Q, M_K\} \in \mathbb{R}^{hw}$  represent the foreground regions of pairs  $\{Q, K\}$ . Under three different settings introduced in Problem Definition, the Weights Generator of WAM produces different weighted maps  $\{M_Q, M_K\}$  as follows:

- **GTBoxR** setting. We denote  $\{M_t, M_r\} \in \mathbb{R}^{hw}$  as the foreground regions of  $\{C_3^t, C_3^r\}$ . In the case of implementing  $\bar{C}_3^t = f_{WAM}(C_3^t, C_3^r)$ , there are  $M_Q = M_t$  and  $M_K = M_r$ . Calculating  $\bar{C}_3^r = f_{WAM}(C_3^r, C_3^t)$  assigns  $M_Q = M_r$  and  $M_K = M_t$ . The following assignment rules of  $\{M_Q, M_K\}$  with  $\{M_t, M_r\}$  are the same, thus we only introduce how to generate  $\{M_t, M_r\}$  for simplicity. (a) This setting gives ground-truth  $\{B_i^t\}$  for reference image, then we generate  $M_r$  by assigning the  $\{B_i^t\}$  regions as  $1 + \alpha_1$  and other background regions as 1. (b) For target feature  $C_3^t$ , we first use a light decoder to predict a semantic segmentation mask  $m_t$ . Then we produce  $M_t$  by assigning the foreground regions of  $m_t$  that are higher than 0.5 as  $1 + \alpha_1$ , and other background regions as 1. The detail of the light segmentation decoder is described in the following.

- **PreBoxR** setting. Similarly, we use the predicted  $\{\hat{B}_i^r\}$  and  $m_t$  to generate  $M_r$  and  $M_t$ , respectively.

- **NoBoxR** setting. The light decoder predicts two semantic segmentation mask  $m_r$  and  $m_t$  to generate  $M_r$  and  $M_t$ .

The light segmentation decoder consists of two  $3 \times 3$  convolution layers and a  $1 \times 1$  convolution layer, which per-pixel classifies the foreground and background of the image. The light segmentation decoder is optimized by the box-supervised segmentation loss proposed in BoxInst (Tian et al. 2021), which utilizes the ground-truth bounding boxes to supervise the segmentation decoder. Fig.4 presents the generated Weighted Map as the foreground regions.

There are two important abilities of WAM: (a) The features  $\{\bar{C}_3^t, \bar{C}_3^r\}$  processed through WAM are more discriminative than  $\{C_3^t, C_3^r\}$ . The dense local features  $\{C_3^t, C_3^r\}$  extracted by Convolutional Neural Networks (CNNs) have limited receptive field which may not distinguish indistinctive or repetitive regions. Transforming by WAM, the global features  $\{\bar{C}_3^t, \bar{C}_3^r\}$  have larger global context to find the correspondences among surrounding regions. (b) The WAM enhances the connection between foreground regions with the constrained of  $\{M_Q, M_K\}$ , which benefits the subsequent matching layer to obtain high-quality matches. The Cross

Attention in Transformer learns global context based on the affinity matrix between feature pairs, which may easily be disturbed by background noise and leads to false matches. With the constrained of  $\{M_Q, M_K\}$ , the WAM can be more focus on interacting information among the foreground regions to ensure high-quality matches.

**Weighted Spatial Attention Module** The Weighted Spatial Attention Module  $f_{WSAM}$  highlights the foreground regions of target image, which is expressed as:

$$C_3^{t'} = f_{WSAM}(C_3^t, C_3^r) \quad (5)$$

The WSAM stacks a Weighted Spatial Attention block and a Self-Attention block. And the structure of the key component Weighted Spatial Attention (WSAttention) is illustrated in Fig.3(c), which is formulated as:

$$\begin{aligned} Q' &= \text{WSAttention}((Q, K, V), M_Q, M_K) \\ &= Q \odot (1 + M_{QV}) \end{aligned} \quad (6)$$

$$\text{where, } M_{QV}(i) = \langle Q(i), V^{\sim Q}(i) \rangle, \quad (6)$$

$$V^{\sim Q} = \text{WeightedAttention}((Q, K, V), M_Q, M_K)$$

Here we explain the meanings of key variables in Weighted Spatial Attention: (a) The  $V^{\sim Q}$ , is  $Q$ -aligned  $V$ , i.e. aggregates  $V$  into alignment with  $Q$ . At  $i^{\text{th}}$  spatial position,  $V^{\sim Q}(i)$  is aggregated from  $V$  with the affinity vector between  $(Q(i), K)$ . Intuitively,  $V^{\sim Q}(i)$  collects all the patch features from  $V$  that are semantically similar to  $Q(i)$ . (b)  $Q'$ , is  $M_{QV}$ -enhanced  $Q$ , i.e. uses the cosine similarity map  $M_{QV}$  between  $(Q, V^{\sim Q})$  to re-weight  $Q$ . Therefore,  $Q'$  will be enhanced if  $Q$  is similar to  $V^{\sim Q}$  (i.e.  $Q$ -aligned  $V$ ). Thus Weighted Spatial Attention is able to enhance the spatial response of  $Q$  via the similar regions with  $V$ .

Based on the above analysis, we propose to use instance feature  $C_3^r$  and learnable semantic embedding  $W_e \in \mathbb{R}^{C \times c_3}$  (i.e. pytorch code is  $W_e = \text{nn.Embedding}(C, c_3)$ ) to highlight  $C_3^t$ , i.e.  $C_3^t$  can be highlighted by the similar regions of  $C_3^r$  and  $W_e$ , where  $W_e$  contains foreground semantics via learning all categories information (Lai et al. 2022). Formally,

$$\begin{aligned} Q' &= \text{WSAttention}((C_3^t, C_3^r, C_3^r), M_Q, M_K) \\ &+ \text{WSAttention}(C_3^t, W_e, W_e) \end{aligned} \quad (7)$$

The WSAM needs to find the foreground regions of  $C_3^r$  to highlight  $C_3^t$ . Under different settings, the Weights Generator of WSAM produces different weighted maps  $\{M_Q, M_K\} = \{M_t, M_r\}$  (i.e. the potential foreground regions of  $\{C_3^t, C_3^r\}$ ) as follows:

- **GTBoxR** setting. For  $C_3^r$ , we generate  $M_r$  by assigning the given ground-truth  $\{B_i^r\}$  regions as  $1 + \alpha_2$  and other background regions as 1. For  $C_3^t$ , we use the estimated homography  $\mathcal{H}' = f_M(f_{WAM}(f_B(x^t, x^r)))$  to affine  $M_r$  to generate  $M_t = \mathcal{H}'M_r$ .

- **PreBoxR** setting. Similarly, we use the predicted  $\{\hat{B}_i^r\}$  to generate  $M_r$ , and then obtains  $M_t = \mathcal{H}'M_r$ .

- **NoBoxR** setting. Firstly, the light decoder predicts two semantic segmentation mask  $m_r$  and  $m_t$  to generate  $M_r$  and  $M_t$ , respectively. Then we further refine the maps by  $M_r = M_r + \mathcal{H}'^{-1}M_t$  and  $M_t = M_t + \mathcal{H}'M_r$ .

**Match Head with Box Filter** **Box Filter**  $f_{BF}$  produces an filter map  $\mathcal{F}$  to mitigate the impact of false matches measured by  $\mathcal{H}' = f_M(f_{WAM}(f_B(x^t, x^r)))$  on stage ②. After stage ③, Detector branch outputs the prediction boxes  $\{\hat{B}_i^t\}$ . On stage ④, we first generate foreground regions  $\hat{M}_t$  of target image by assigning  $\{\hat{B}_i^t\}$  regions as  $1 + \beta$  and other background regions as 1, and then we obtain the foreground  $\hat{M}_r$  of reference image from WSAM with  $M_r$  of which foreground is also re-assigned as  $1 + \beta$ . Finally, filter map  $\mathcal{F}$  is generated by  $\mathcal{F}(i, j) = \hat{M}_t(i) \cdot \hat{M}_r(j)$ .

**Match Head**  $f_M$  is based on the dual-softmax matching layer (Tyszkiewicz, Fua, and Trulls 2020; Sun et al. 2021). Formally, the matching probability  $\mathcal{P}$  is obtained by:

$$\begin{aligned} \mathcal{P}(i, j) &= \sigma(\mathcal{S}(i, \cdot))_j \cdot \sigma(\mathcal{S}(\cdot, j))_i, \\ \text{where, } \mathcal{S}(i, j) &= \frac{1}{\tau} \cdot (\bar{C}_3^t(i), \bar{C}_3^r(j)) \end{aligned} \quad (8)$$

where  $\mathcal{S}$  is the score matrix between the feature pairs  $(\bar{C}_3^t, \bar{C}_3^r)$ . The dual-softmax matching applies softmax on both dimensions of  $\mathcal{S}$  to obtain the matching probability  $\mathcal{P}$ . Then, we use the filter map  $\mathcal{F}$  generated by Box Filter to update  $\mathcal{P}$  by  $\mathcal{P}(i, j) \leftarrow \mathcal{P}(i, j) \cdot \mathcal{F}(i, j)$ . Next, we obtain the match prediction by  $\hat{\mathcal{M}} = \{(\tilde{i}, \tilde{j}) \mid \forall (\tilde{i}, \tilde{j}) \in \text{MNN}(\mathcal{P}), \mathcal{P}(\tilde{i}, \tilde{j}) \geq \theta\}$ , where, MNN is a Mutual Nearest Neighbor operator, and  $\theta$  is a threshold to select good matches. Finally, the homography matrix is estimate by RANSAC algorithm with  $\hat{\mathcal{H}} = \text{RANSAC}(\hat{\mathcal{M}})$ .

**Loss Function** The MatchDet network is optimized by the loss function:  $\mathcal{L} = \mathcal{L}_{matcher} + \lambda \mathcal{L}_{detector}$ , where  $\mathcal{L}_{matcher}$  and  $\mathcal{L}_{detector}$  are the losses of Matcher branch and Detector branch respectively, and  $\lambda$  is the balance weight. The  $\mathcal{L}_{detector}$  is the same as FCOS (Tian et al. 2019). Following LoFTR (Sun et al. 2021), the  $\mathcal{L}_{matcher}$  is formulated as:  $\mathcal{L}_{matcher} = -\frac{1}{|\mathcal{M}|} \sum_{(\tilde{i}, \tilde{j}) \in \mathcal{M}} \log \hat{\mathcal{M}}(\tilde{i}, \tilde{j})$ , where  $\mathcal{M}$  and  $\hat{\mathcal{M}}$  are ground-truth and predicted matches.

## Discussion

**Task-collaborative vs. Task-individual** Comparing to the current task-individual solutions, the advantages of our task-collaborative framework MatchDet are: (a) With the collaborative learning of image matching and object detection tasks, MatchDet is able to obtain mutual performance improvements. (b) Lower complexity with a shared backbone. (c) More convenient in practical applications with one single MatchDet model for two tasks.

**WSAM vs. WAM** The WAM transforms the pair features of target and reference images into new feature space that are easy to measure similarity for matching, while the WSAM utilizes a spatial map to enhance the foregrounds of target image for detection. Details are presented in APPENDIX.

**Weighted Attention vs. Masked Attention** The proposed Weighted Attention has three main differences from Masked Attention (Cheng et al. 2022), including input forms, weighting operation, and weights generation. These

{Target image, Reference image}	Method	Average Precision on Target image						Homography est. AUC			Params	FLOPs
		AP	AP <sub>50</sub>	AP <sub>75</sub>	AP <sub>S</sub>	AP <sub>M</sub>	AP <sub>L</sub>	@3px	@5px	@10px		
{Warp-COCO, COCO}	FCOS (Tian et al. 2019)	31.02	47.99	33.08	11.09	28.10	43.66	-	-	-	32.29M	174604M
	LoFTR (Sun et al. 2021)	-	-	-	-	-	-	47.94	68.15	84.02	27.67M	171076M
	<b>DBase</b>	30.94	47.65	32.66	10.82	27.76	43.32	-	-	-	32.29M	174604M
	<b>MBase</b>	-	-	-	-	-	-	38.94	58.67	78.39	23.51M	143340M
	<b>MDBase</b>	30.95	47.57	32.67	11.23	27.50	43.48	36.42	56.78	77.32	32.29M	206641M
	<b>MatchDet-G</b>	<b>43.62</b>	<b>61.66</b>	<b>47.32</b>	<b>19.71</b>	<b>41.90</b>	<b>56.74</b>	<b>63.18</b>	<b>77.86</b>	<b>89.02</b>	38.54M	248246M
	<b>MatchDet-P</b>	34.15	51.23	35.98	14.79	31.19	46.13	58.31	73.54	83.24	38.54M	248246M
<b>MatchDet-N</b>	32.28	49.28	34.01	12.81	29.40	44.38	54.11	69.06	79.82	38.54M	257154M	
{miniScanNet-F0, miniScanNet-F1}	FCOS (Tian et al. 2019)	25.13	39.56	25.63	0.26	4.43	29.67	-	-	-	32.15M	60709M
	LoFTR (Sun et al. 2021)	-	-	-	-	-	-	10.75	31.48	50.02	27.67M	69729M
	<b>DBase</b>	24.99	39.22	25.50	0.24	4.39	29.52	-	-	-	32.15M	60709M
	<b>MBase</b>	-	-	-	-	-	-	5.87	16.34	33.66	23.51M	50590M
	<b>MDBase</b>	23.70	38.05	23.60	0.25	3.46	28.15	5.95	16.50	33.93	32.15M	72016M
	<b>MatchDet-G</b>	<b>40.69</b>	<b>58.12</b>	<b>41.38</b>	<b>1.88</b>	<b>25.76</b>	<b>45.03</b>	<b>18.43</b>	<b>39.14</b>	<b>57.11</b>	38.39M	100725M
	<b>MatchDet-P</b>	28.29	43.12	29.34	0.82	6.54	33.03	14.39	35.18	53.24	38.39M	100725M
<b>MatchDet-N</b>	27.33	42.09	27.58	0.56	5.47	32.29	12.47	33.09	51.32	38.39M	104289M	

Table 1: MatchDet vs. other methods on different combinations of {Target image, Reference image} pairs on Full Warp-COCO and miniScanNet datasets. The *Average Precision* on Target image, homography estimation AUC of the corner error in percentage, epochs=12, params and FLOPs are reported. The DBase = [ $f_B, f_F, f_D$ ] and MBase = [ $f_B, f_M$ ] are detector network and matcher network split from MDBase, respectively. The MatchDet-G, MatchDet-P and MatchDet-N follow three different settings of **GTBoxR**, **PreBoxR** and **NoBoxR**, respectively.

Method	Ave. Precision			Homo. est. AUC		
	AP	AP <sub>50</sub>	AP <sub>75</sub>	@3px	@5px	@10px
LoFTR	-	-	-	45.88	65.15	84.99
FCOS	37.32	55.69	40.37	-	-	-
<b>DBase(FC)</b>	37.19	55.54	39.96	-	-	-
<b>MBase</b>	-	-	-	34.86	54.25	74.91
<b>MDBase(FC)</b>	37.07	55.38	39.68	32.67	52.76	73.24
<b>MatchDet-G(FC)</b>	<b>43.99</b>	<b>61.93</b>	<b>47.75</b>	<b>60.09</b>	<b>74.58</b>	<b>86.07</b>
<b>MatchDet-P(FC)</b>	40.57	58.66	42.39	55.21	70.32	80.17
<b>MatchDet-N(FC)</b>	39.49	57.47	41.25	51.24	66.12	76.79
Faster-RCNN	35.90	56.28	39.05	-	-	-
<b>MatchDet-G(FR)</b>	<b>42.22</b>	<b>62.44</b>	<b>45.98</b>	<b>68.72</b>	<b>80.83</b>	<b>90.37</b>
<b>MatchDet-P(FR)</b>	39.79	60.13	43.29	65.13	79.02	87.56
<b>MatchDet-N(FR)</b>	37.85	58.72	41.55	62.37	76.43	85.68
AdaMixer	40.82	59.72	43.73	-	-	-
<b>MatchDet-G(AM)</b>	<b>48.19</b>	<b>66.57</b>	<b>52.35</b>	<b>68.88</b>	<b>80.96</b>	<b>90.96</b>
<b>MatchDet-P(AM)</b>	45.01	64.35	49.23	65.21	78.53	88.55
<b>MatchDet-N(AM)</b>	42.19	62.52	46.22	62.74	76.69	86.88

Table 2: The results of our MatchDet with different detectors, on {COCO, Warp-COCO}. The FC, FR, and AM represent FCOS, Faster-RCNN, and AdaMixer, respectively.

differences make Masked Attention can not be directly integrated by our MatchDet. Masked Attention performs cross-attention between input features and learnable embedding, while Weighted Attention processes a pair of features. Due to this difference in input forms, Weighted Attention needs to generate two corresponding weighted maps for feature pairs. Besides, Mask2Former is applied in segmentation task with the supervision of ground-truth masks, which is easy to obtain good quality masks for Masked Attention. While our MatchDet is proposed for detection task with ground-truth bounding boxes only, thus we propose a Weights Generator using the weakly-supervised segmentation technique to

Method	Modules			Ave. Precision		Homo. est. AUC		
	A.	S.	B.	AP	AP <sub>50</sub>	@3px	@5px	@10px
<b>MBase</b>	-	-	-	30.95	47.57	36.42	56.78	77.32
<b>MatchDet-G</b>	✓	-	-	30.66	47.25	60.12	74.92	87.01
<b>MatchDet-G</b>	✓	✓	-	<b>43.62</b>	<b>61.66</b>	61.15	75.95	87.92
<b>MatchDet-G</b>	✓	✓	✓	<b>43.62</b>	<b>61.66</b>	<b>63.18</b>	<b>77.86</b>	<b>89.02</b>
<b>MatchDet-P</b>	✓	-	-	30.69	47.27	55.43	70.49	81.54
<b>MatchDet-P</b>	✓	✓	-	<b>34.15</b>	<b>51.23</b>	56.42	71.48	82.35
<b>MatchDet-P</b>	✓	✓	✓	<b>34.15</b>	<b>51.23</b>	<b>58.31</b>	<b>73.54</b>	<b>83.24</b>
<b>MatchDet-N</b>	✓	-	-	30.60	47.21	51.67	66.52	78.20
<b>MatchDet-N</b>	✓	✓	-	<b>32.28</b>	<b>49.28</b>	52.43	67.49	79.02
<b>MatchDet-N</b>	✓	✓	✓	<b>32.28</b>	<b>49.28</b>	<b>54.11</b>	<b>69.06</b>	<b>79.82</b>

Table 3: The influence of WAM (A.), WSAM (S.) and Box Filter (B.) on {Warp-COCO, COCO} pairs dataset.

generate coarse masks for Weighted Attention.

## Experiments

### Datasets

**Warp-COCO** The Full Warp-COCO dataset consists of Warp-COCO and COCO (Lin et al. 2014), i.e. every image in COCO has a corresponding transformed synthetic image to make a pair, which has ground-truth poses and boxes.

**miniScanNet** We selected 20 categories from ScanNet (Dai et al. 2017) dataset (230M image pairs) with high-quality bounding boxes to make a new miniScanNet dataset (188K image pairs), which is 1000 times smaller than ScanNet. The miniScanNet consists of real-world image pairs, which is more challenging than the synthetic Warp-COCO dataset.

### Evaluation Metrics and Implementation Details

The *Average Precision* (AP) and the Area Under the Cumulative curve (AUC) of the corner error, are used for object

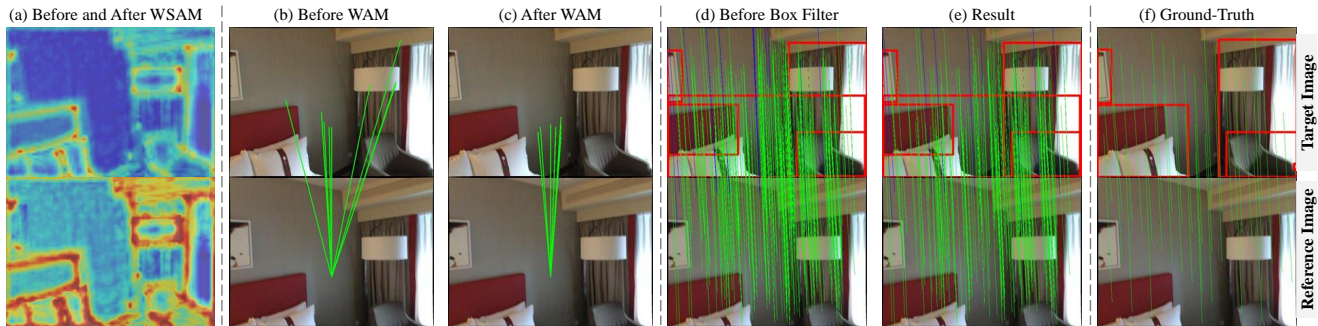


Figure 5: The visualizations for WAM, WSAM, Box Filter and MatchDet results under **GTBoxR** setting from miniScanNet. (a) - (e), are the results processed by the corresponding modules before and after, respectively. (e) shows the predicted bounding boxes and matching results of MatchDet, where these matches are obtained after Box Filter. (f) is Ground-Truth.

$f_{WAM}$	$f_{WSAM}$	Average Precision			Homography est. AUC		
		AP	AP <sub>50</sub>	AP <sub>75</sub>	@3px	@5px	@10px
C.A.	C.A.	31.01	47.72	32.91	45.84	66.13	82.89
W.A.	W.A.	40.41	58.43	44.21	60.03	74.88	86.99
W.S.A.	W.S.A.	42.13	60.07	46.00	58.46	73.40	86.08
W.S.A.	W.A.	40.02	58.00	43.86	57.88	72.89	85.66
W.A.	W.S.A.	<b>43.62</b>	<b>61.66</b>	<b>47.32</b>	<b>61.15</b>	<b>75.95</b>	<b>87.92</b>

Table 4: The influence of Weighted Attention (W.A.) and Weighted Spatial Attention (W.S.A.), under **GTBoxR** setting on {Warp-COCO, COCO} pairs dataset. The  $f_{WAM}$  and  $f_{WSAM}$  are inserted into Matcher and Detector respectively, i.e.  $f_{WAM}$  and  $f_{WSAM}$  directly affect the AUC and the AP respectively. We give different combinations of Cross Attention (C.A.), W.A. and W.S.A.

detection and image matching, respectively. The used backbone is ResNet-50 (He et al. 2016). The data augmentation is not applied for all methods. The hyper-parameters are set as  $\alpha_1 = 1.0$ ,  $\alpha_2 = 1.0$ ,  $\beta = 1.0$ ,  $\lambda = 1.0$ .

### Comparison with Related Methods

As shown in Tab.1 and Tab.2, we conduct experiments on Full Warp-COCO and miniScanNet datasets with different combinations of {Target image, Reference image} pairs, including {Warp-COCO, COCO}, {COCO, Warp-COCO} and {miniScanNet-F0, miniScanNet-F1}. The DBase and MBase are the two task-individual baselines for object detection and image matching, respectively. The MDBase is the task-collaborative baseline for Match-and-Detection task, which obtains similar AP and AUC compared to DBase and MBase respectively. Under three different settings of **GTBoxR**, **PreBoxR** and **NoBoxR**, our MatchDet outperforms these baselines on both AP and AUC performances, which demonstrates that the proposed approaches can achieve the collaborative learning between matching and detection tasks to obtain mutual performance improvements. Under the **GTBoxR** setting, MatchDet obtains a larger performance improvement. Even under the most challenging **NoBoxR** setting, our MatchDet still achieves competitive improvement.

### Ablation Study

**WAM, WSAM and Box Filter.** The results in Tab.3 demonstrate that the proposed three modules achieve consistent improvements under the settings of **GTBoxR**, **Pre-BoxR** and **NoBoxR**. Specifically, the WSAM highlights the foreground regions of target image to boost the AP of the foregion task. The WAM and Box Filter obtains high-quality matches to improve the AUC of image matching task.

**Weighted Attention and Weighted Spatial Attention.** The results in Tab.4 show that: (a) The best setting is using Weighted Attention for Matcher and Weighted Spatial Attention for Detector. These experimental results coincide with the theoretical analysis in APPENDIX. (b) The proposed Weighted Attention and Weighted Spatial Attention, exploring the correlation between foreground regions of feature pairs, are able to obtain better performance than the traditional Cross Attention.

### Visualization Analysis

Fig.5 shows the visualizations for WAM, WSAM, Box Filter and MatchDet results under **GTBoxR** setting. Fig.5(a) shows that the WSAM is able to highlight the foreground regions. Comparing Fig.5(c) to Fig.5(b), the WAM finds the correspondences among surrounding regions to produce more discriminative feature representations. Comparing Fig.5(e) to Fig.5(d), Box Filter reduces the background interference to obtain high-quality matches.

### Conclusion

In this paper, we propose a collaborative framework called MatchDet for image matching and object detection to obtain mutual improvements. To achieve the collaborative learning of the two tasks, three novel modules are proposed, including a Weighted Spatial Attention Module (WSAM) which highlights the foreground regions of target image for Detector, and Weighted Attention Module (WAM) and Box Filter which obtains high-quality matches for Matcher. The experimental results on Warp-COCO and miniScanNet datasets show that our approaches are effective and achieve competitive improvements.

## References

- Bay, H.; Tuytelaars, T.; and Van Gool, L. 2006. Surf: Speeded up robust features. In *European conference on computer vision*, 404–417. Springer.
- Cai, Z.; and Vasconcelos, N. 2018. Cascade r-cnn: Delving into high quality object detection. In *CVPR*.
- Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-end object detection with transformers. In *ECCV*.
- Chen, H.; Luo, Z.; Zhou, L.; Tian, Y.; Zhen, M.; Fang, T.; McKinnon, D.; Tsin, Y.; and Quan, L. 2022. ASpanFormer: Detector-Free Image Matching with Adaptive Span Transformer. In *ECCV*.
- Cheng, B.; Misra, I.; Schwing, A. G.; Kirillov, A.; and Girdhar, R. 2022. Masked-attention mask transformer for universal image segmentation. In *CVPR*.
- Christoph, F.; Axel, P.; and Andrew, Z. 2017. Detect to track and track to detect. In *ICCV*.
- Dai, A.; Chang, A. X.; Savva, M.; Halber, M.; Funkhouser, T.; and Nießner, M. 2017. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5828–5839.
- Dalal, N.; and Triggs, B. 2005. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, 886–893. Ieee.
- Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *CVPR*.
- DeTone, D.; Malisiewicz, T.; and Rabinovich, A. 2018. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 224–236.
- Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2020. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.
- Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*.
- Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; and Tian, Q. 2019. Centernet: Keypoint triplets for object detection. In *ICCV*.
- Dusmanu, M.; Rocco, I.; Pajdla, T.; Pollefeys, M.; Sivic, J.; Torii, A.; and Sattler, T. 2019. D2-Net: A Trainable CNN for Joint Detection and Description of Local Features. In *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- Gao, Z.; Wang, L.; Han, B.; and Guo, S. 2022. Adamixer: A fast-converging query-based object detector. In *CVPR*.
- Harris, C.; Stephens, M.; et al. 1988. A combined corner and edge detector. In *Alvey vision conference*, 10–5244. Citeseer.
- He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask r-cnn. In *ICCV*.
- He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*.
- Huang, L.; Yang, Y.; Deng, Y.; and Yu, Y. 2015. Densebox: Unifying landmark localization with end to end object detection. *arXiv preprint arXiv:1509.04874*.
- Kong, T.; Sun, F.; Liu, H.; Jiang, Y.; Li, L.; and Shi, J. 2020. FoveaBox: Beyond Anchor-based Object Detection. *IEEE Transactions on Image Processing*.
- Lai, J.; Yang, S.; Liu, W.; Zeng, Y.; Huang, Z.; Wu, W.; Liu, J.; Gao, B.-B.; and Wang, C. 2022. tSF: Transformer-Based Semantic Filter for Few-Shot Learning. In *ECCV*.
- Law, H.; and Deng, J. 2018. Cornernet: Detecting objects as paired keypoints. In *ECCV*.
- Li, F.; Zhang, H.; Liu, S.; Guo, J.; Ni, L. M.; and Zhang, L. 2022. Dn-detr: Accelerate detr training by introducing query denoising. In *CVPR*.
- Liang, J.; Homayounfar, N.; Ma, W.-C.; Xiong, Y.; Hu, R.; and Urtasun, R. 2020. Polytransform: Deep polygon transformer for instance segmentation. In *CVPR*.
- Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; and Belongie, S. 2017a. Feature pyramid networks for object detection. In *CVPR*.
- Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollár, P. 2017b. Focal loss for dense object detection. In *ICCV*.
- Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *ECCV*.
- Liu, L.; Ouyang, W.; Wang, X.; Fieguth, P.; Chen, J.; Liu, X.; and Pietikäinen, M. 2020. Deep learning for generic object detection: A survey. *International journal of computer vision*, 128(2): 261–318.
- Lowe, D. G. 2004. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2): 91–110.
- Luo, Z.; Shen, T.; Zhou, L.; Zhang, J.; Yao, Y.; Li, S.; Fang, T.; and Quan, L. 2019. Contextdesc: Local descriptor augmentation with cross-modality context. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2527–2536.
- Moravec, H. P. 1981. Rover Visual Obstacle Avoidance. In *IJCAI*, volume 81, 785–790.
- Mur-Artal, R.; Montiel, J. M. M.; and Tardos, J. D. 2015. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE transactions on robotics*, 31(5): 1147–1163.
- Nebahay, G.; and Pflugfelder, R. 2014. Consensus-based matching and tracking of keypoints for object tracking. In *WACV*.
- Redmon, J.; and Farhadi, A. 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.
- Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015a. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*.
- Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015b. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*.



- Revaud, J.; De Souza, C.; Humenberger, M.; and Weinzaepfel, P. 2019. R2d2: Reliable and repeatable detector and descriptor. *Advances in neural information processing systems*, 32: 12405–12415.
- Rublee, E.; Rabaud, V.; Konolige, K.; and Bradski, G. 2011. ORB: An efficient alternative to SIFT or SURF. In *2011 International conference on computer vision*, 2564–2571. Ieee.
- Sarlin, P.-E.; DeTone, D.; Malisiewicz, T.; and Rabinovich, A. 2020. SuperGlue: Learning feature matching with graph neural networks. In *CVPR*.
- Schmid, C.; and Mohr, R. 1997. Local grayvalue invariants for image retrieval. *IEEE transactions on pattern analysis and machine intelligence*, 19(5): 530–535.
- Shrivastava, A.; Malisiewicz, T.; Gupta, A.; and Efros, A. A. 2011. Data-driven visual similarity for cross-domain image matching. In *Proceedings of the 2011 SIGGRAPH Asia Conference*, 1–10.
- Sun, J.; Shen, Z.; Wang, Y.; Bao, H.; and Zhou, X. 2021. LoFTR: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8922–8931.
- Tian, Z.; Shen, C.; Chen, H.; and He, T. 2019. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9627–9636.
- Tian, Z.; Shen, C.; Wang, X.; and Chen, H. 2021. Boxinst: High-performance instance segmentation with box annotations. In *CVPR*.
- Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; and Jégou, H. 2021. Training data-efficient image transformers & distillation through attention. In *ICML*.
- Tyszkiewicz, M. J.; Fua, P.; and Trulls, E. 2020. DISK: Learning local features with policy gradient. In *NeurIPS*.
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *NeurIPS*.
- Wang, J.; Chen, K.; Yang, S.; Loy, C. C.; and Lin, D. 2019. Region Proposal by Guided Anchoring. In *CVPR*.
- Wang, Q.; Zhang, J.; Yang, K.; Peng, K.; and Stiefelhagen, R. 2022. MatchFormer: Interleaving Attention in Transformers for Feature Matching. In *ACCV*.
- Wang, Q.; Zhou, X.; Hariharan, B.; and Snavely, N. 2020. Learning Feature Descriptors using Camera Pose Supervision. In *Proc. European Conference on Computer Vision (ECCV)*.
- Wang, W.; Xie, E.; Li, X.; Fan, D.-P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; and Shao, L. 2021. Pyramid Vision Transformer: A Versatile Backbone for Dense Prediction without Convolutions. In *ICCV*.
- Xie, E.; Wang, W.; Yu, Z.; Anandkumar, A.; Alvarez, J. M.; and Luo, P. 2021. SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers. In *NeurIPS*.
- Yang, Z.; Liu, S.; Hu, H.; Wang, L.; and Lin, S. 2019. Reppoints: Point set representation for object detection. In *ICCV*.
- Yi, K. M.; Trulls, E.; Lepetit, V.; and Fua, P. 2016. Lift: Learned invariant feature transform. In *European conference on computer vision*, 467–483. Springer.
- Zeng, A.; Song, S.; Yu, K.-T.; Donlon, E.; Hogan, F. R.; Bauza, M.; Ma, D.; Taylor, O.; Liu, M.; Romo, E.; Fazeli, N.; Alet, F.; Daffe, N. C.; Holladay, R.; Morona, I.; Nair, P. Q.; Green, D.; Taylor, I.; Liu, W.; Funkhouser, T.; and Rodriguez, A. 2018. Robotic Pick-and-Place of Novel Objects in Clutter with Multi-Affordance Grasping and Cross-Domain Image Matching. In *ICRA*.
- Zhang, D.; Zhang, H.; Tang, J.; Wang, M.; Hua, X.; and Sun, Q. 2020a. Feature pyramid transformer. In *ECCV*.
- Zhang, H.; Li, F.; Liu, S.; Zhang, L.; Su, H.; Zhu, J.; Ni, L. M.; and Shum, H.-Y. 2023. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *ICLR*.
- Zhang, S.; Chi, C.; Yao, Y.; Lei, Z.; and Li, S. Z. 2020b. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *CVPR*.
- Zhang, X.; Wan, F.; Liu, C.; Ji, R.; and Ye, Q. 2019. FreeAnchor: Learning to match anchors for visual object detection. In *NIPS*.
- Zhang, Z.; Deriche, R.; Faugeras, O.; and Luong, Q.-T. 1995. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial intelligence*, 78(1-2): 87–119.
- Zheng, S.; Lu, J.; Zhao, H.; Zhu, X.; Luo, Z.; Wang, Y.; Fu, Y.; Feng, J.; Xiang, T.; Torr, P. H.; et al. 2021. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *CVPR*.
- Zhou, X.; Wang, D.; and Krähenbühl, P. 2019. Objects as points. *arXiv preprint arXiv:1904.07850*.
- Zhu, C.; He, Y.; and Savvides, M. 2019. Feature selective anchor-free module for single-shot object detection. In *CVPR*.
- Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; and Dai, J. 2020. Deformable DETR: Deformable Transformers for End-to-End Object Detection. In *ICLR*.

## APPENDIX

### Related Work

**Object Detection** The current object detection algorithms can be divided into anchor-based, anchor-free, and query-based methods. The anchor-based two-stage (Ren et al. 2015a; He et al. 2017) and multi-stage (Cai and Vasconcelos 2018) methods utilize anchors to produce proposals for classification and box regression, and the anchor-based one-stage methods (Redmon and Farhadi 2018; Lin et al. 2017b; Zhang et al. 2019, 2020b; Wang et al. 2019) can directly classify and regress anchor boxes without relying on object proposal. The anchor-free methods (Law and Deng 2018; Duan et al. 2019; Zhou, Wang, and Krähenbühl 2019; Yang et al. 2019) implement object detection via detecting key-point or semantic-point, and another anchor-free methods (Huang et al. 2015; Zhu, He, and Savvides 2019; Kong et al. 2020; Tian et al. 2019) obtain the detection results by

densely classify each point on feature pyramids (Lin et al. 2017a), and then predict the distances between the point and the four sides of box. Recently, query-based detector, DETR (Carion et al. 2020), predicts a set of objects by attending queries to the feature map with the transformer decoder and achieves promising performance. Since the introduction of DETR, there have been several modifications and improvements (Zhu et al. 2020; Li et al. 2022; Gao et al. 2022; Zhang et al. 2023) proposed to overcome its limitations, such as slow training convergence and performance drops for small objects. In this paper, we choose the classical anchor-free FCOS (Tian et al. 2019) as the basic detector due to its good performance and simplicity.

**Image Matching** Image matching finds pixel-wise correspondences between pairs of images. Early works focused on designing interest point detectors and descriptors (Moravec 1981; Harris, Stephens et al. 1988; Zhang et al. 1995; Schmid and Mohr 1997; Lowe 2004; Dalal and Triggs 2005; Bay, Tuytelaars, and Van Gool 2006; Rublee et al. 2011). Recently, learned interest point detectors and descriptors have been increasingly popular (Yi et al. 2016; DeTone, Malisiewicz, and Rabinovich 2018; Dusmanu et al. 2019; Revaud et al. 2019; Luo et al. 2019; Tyszkiewicz, Fua, and Trulls 2020; Wang et al. 2020), typically outperforming the hand-crafted counterparts. More recently, the interest point detector-free method, LoFTR (Sun et al. 2021), finds the correspondences by matching on dense and representative feature maps, without detecting keypoints. LoFTR (Sun et al. 2021) and the follow-ups (Wang et al. 2022; Chen et al. 2022) match points distributed on dense grids rather than sparse locations, which boosts the robustness to impressive levels. MatchFormer (Wang et al. 2022) performs feature extraction and similarity learning through a transformer synchronously, which can provide matching-aware features in each stage of the hierarchical structure. To capture both global context and local details, ASpanFormer (Chen et al. 2022) proposes a Transformer-based detector-free matcher, equipped with a hierarchical attention framework. In this paper, we choose the classical LoFTR (Sun et al. 2021) as the basic image matching method due to its good performance and simplicity.

**Transformer Attention** Transformer is an attention-based architecture which is firstly introduced in natural language processing (Vaswani et al. 2017; Devlin et al. 2020). Due to its powerful ability in learning representation, Transformer has been widely applied in vision tasks like image classification (Dosovitskiy et al. 2021; Wang et al. 2021; Touvron et al. 2021), segmentation (Zheng et al. 2021; Liang et al. 2020; Xie et al. 2021), object detection (Zhang et al. 2020a; Carion et al. 2020; Zhu et al. 2020) and image matching (Sun et al. 2021; Sarlin et al. 2020). The Cross Attention used in LoFTR (Sun et al. 2021) learns global context based on the full affinity matrix between feature pairs, but it may easily be distracted by background and causes the slow convergence of model. To alleviate this problem, the Masked Attention was proposed in Mask2Former (Cheng et al. 2022), which only made attention within the foreground regions of the predicted mask for query embedding. Inspired by Masked Attention, we propose two novel transformer-based

weighted attention and weighted spatial attention modules to strengthen the target object, by exploring the correlation between the foreground regions of feature pairs.

## Discussion

### WSAM vs. WAM

The WAM transforms the pair features of target and reference images into new feature space that are easy to measure similarity for matching, while the WSAM utilizes a spatial map to enhance the foregrounds of target image for detection. Let’s assume the features consist of foreground and background information. Then we denote  $C_3^t = C_3^t[f] + C_3^t[b]$  and  $C_3^r = C_3^r[f] + C_3^r[b]$ , where  $[f]$  and  $[b]$  represent the foreground and background information, respectively. We assume the similarity scores are  $\langle C_3^t[f], C_3^r[f] \rangle = v \in [0, 1]$  and  $\langle C_3^t[b], C_3^r[b] \rangle = u \in [0, 1]$ , and there is  $v \approx u$  since the pair images are similar. We next discuss the differences of feature processing by WAM and WSAM:

- WAM processing:

$$\begin{aligned} \bar{C}_3^t[f] &\approx C_3^t[f] + v \cdot C_3^r[f] + (1 - v) \cdot C_3^r[b], \\ \bar{C}_3^r[f] &\approx C_3^r[f] + v \cdot C_3^t[f] + (1 - v) \cdot C_3^t[b] \end{aligned} \quad (9)$$

(a) For image matching task, the goal is to calculate the similarity between foreground regions  $\bar{C}_3^t[f]$  and  $\bar{C}_3^r[f]$ . According to Eq.9, there is  $\langle \bar{C}_3^t[f], \bar{C}_3^r[f] \rangle \approx v + v \cdot v + (1 - v) \cdot u \approx 2v$ , thus the similarity between foreground regions are enhanced with two times for Matcher. (b) For object detection task, it aims to classify  $\bar{C}_3^t[f]$ , but the involved background information  $C_3^r[b]$  may disturb the classification. If  $v < 0.5$ , the background disturbance of  $(1 - v) \cdot C_3^r[b]$  is even larger than the foreground enhancement  $v \cdot C_3^r[f]$ .

- WSAM processing:

$$C_3^{t'}[f] \approx C_3^t[f] + C_3^t[f] \cdot \langle C_3^t[f], \bar{C}_3^r[f] \rangle \quad (10)$$

where  $w = \langle C_3^t[f], \bar{C}_3^r[f] \rangle \approx v + v \cdot 1 + (1 - v) \cdot (1 - v) \approx 1 + v^2$ . Thus the foreground  $C_3^t[f]$  is strengthened with around  $v^2$  times.

### MatchDet vs. Keypoint-base Tracking

(1) We have discussed the differences between MatchDet and Tracker in the Introduction section. our MatchDet achieves a large improvement with 39.96% in image matching task compared to Tracker. (2) The framework of keypoint-base tracking (Nebehay and Pflugfelder 2014) is {detect and describe keypoints  $\rightarrow$  match keypoints  $\rightarrow$  compute keypoints’ displacement}, which solves tracking problem by performing keypoints matching, without integrating object detection method. Differently, we propose a collaborative framework MatchDet for image matching and object detection to obtain mutual improvements. In short, keypoint-base tracking is {matching  $\rightarrow$  tracking}, while MatchDet is {matching  $\Rightarrow$  object detection}.

## Experiments

### Datasets

**Warp-COCO** The Warp-COCO dataset consists of synthetic images generated from the large-scale detection

benchmark COCO (Lin et al. 2014) by implementing perspective transformation. Then the Full Warp-COCO dataset consists of Warp-COCO and COCO, i.e. every image in COCO has a corresponding transformed synthetic image to make a pair, which has ground-truth poses and bounding boxes. Similar to the common practice (Ren et al. 2015b; Tian et al. 2019) in detection task, we use the Full Warp-COCO `trainval35k` split (115K image pairs) for training, and `minival` split (5K image pair) for validation. All images are resized to make the height 800 and the width less or equal to 1333.

**miniScanNet** The ScanNet (Dai et al. 2017) dataset is an image matching benchmark, which has 230M image pairs with ground-truth poses and noisy bounding boxes. We selected the samples of 20 categories with high-quality bounding box labels to make a new miniScanNet dataset, which is 1000 times smaller than ScanNet. Specifically, 188K image pairs with overlap between [40%, 80%] are sampled for training, and 1500 testing pairs are used for evaluation. Each image pairs {miniScanNet-F0, miniScanNet-F1} denote as the previous and the next frame images, respectively. All images are resized to  $640 \times 480$ . The miniScanNet consists of real-world image pairs, which is more challenging than the synthetic Warp-COCO dataset.

## Evaluation Metrics

For object detection, consisting with FCOS (Tian et al. 2019), the most commonly metric *Average Precision* (AP) is used, derived from precision and recall. For image matching, following LoFTR (Sun et al. 2021), we report the area under the cumulative curve (AUC) of the corner error up to threshold values of 3, 5, and 10 pixels, respectively. The corner error is computed between the images warped with the estimated  $\hat{H}$  and the ground-truth  $H$  as a correctness identifier as in (Sun et al. 2021).

## Implementation Details

The used backbone is ResNet-50 (He et al. 2016) with the initialized weights pre-trained on ImageNet (Deng et al. 2009). The data augmentation is not applied for all methods, and the total batch size is 16 pair images trained on 8 GPUs up-to 12 epoches. The model is optimized end-to-end with stochastic gradient descent (SGD) with the initial learning rate of 0.01. The learning rate is reduced by a factor of 10 after 8th epoch. The weight decay and momentum are 0.0001 and 0.9, respectively. The default hyper-parameters are set as  $N_{WAM} = 2$ ,  $N_{WSAM} = 1$ ,  $\alpha_1 = 1.0$ ,  $\alpha_2 = 1.0$ ,  $\beta = 1.0$ ,  $\lambda = 1.0$ . And other hyper-parameters are the same as FCOS (Tian et al. 2019) and LoFTR (Sun et al. 2021).

## Ablation Study

**Stacked Numbers of WAM and WSAM.** As illustrated in Tab.5, stacking more WAM is effective for image matching, while adding more WSAM is not much helpful for object detection. The WAM transforms the pair features of target and reference images into new feature space, hence stacking more WAM is useful. However, the WSAM utilizes a spatial

$N_{WAM}$	$N_{WSAM}$	Average Precision			Homography est. AUC		
		AP	AP <sub>50</sub>	AP <sub>75</sub>	@3px	@5px	@10px
1	1	43.44	61.41	47.10	57.93	72.44	84.99
2	2	43.39	61.38	47.00	60.89	75.32	<b>87.96</b>
2	1	<b>43.62</b>	<b>61.66</b>	<b>47.32</b>	<b>61.15</b>	<b>75.95</b>	87.92

Table 5: The influence of stacked module numbers of WAM and WSAM (i.e.  $N_{WAM}$  and  $N_{WSAM}$ , respectively), under **GTBoxR** setting on {Warp-COCO,COCO} pairs dataset.

map to enhance the foregrounds of target image, and stacking more WSAM only increases the scale of the spatial map, which is useless in further highlighting the target.

**Influence of some hyper-parameters.** Influence of some hyper-parameters is shown in Tab.6.

$\alpha_1$	$\alpha_2$	$\beta$	$\lambda$	Average Precision			Homography est. AUC		
				AP	AP <sub>50</sub>	AP <sub>75</sub>	@3px	@5px	@10px
0.5	0.5	0.5	0.5	41.03	59.05	45.01	<b>63.24</b>	<b>77.63</b>	<b>88.95</b>
0.5	0.5	0.5	1.0	42.59	60.53	46.21	61.92	76.13	88.00
1.0	1.0	1.0	0.5	42.34	60.22	46.03	62.07	76.89	88.33
1.0	1.0	1.0	1.0	<b>43.62</b>	<b>61.66</b>	<b>47.32</b>	61.15	75.95	87.92

Table 6: The influence of some hyper-parameters, under **GT-BoxR** setting on {Warp-COCO,COCO} pairs dataset.

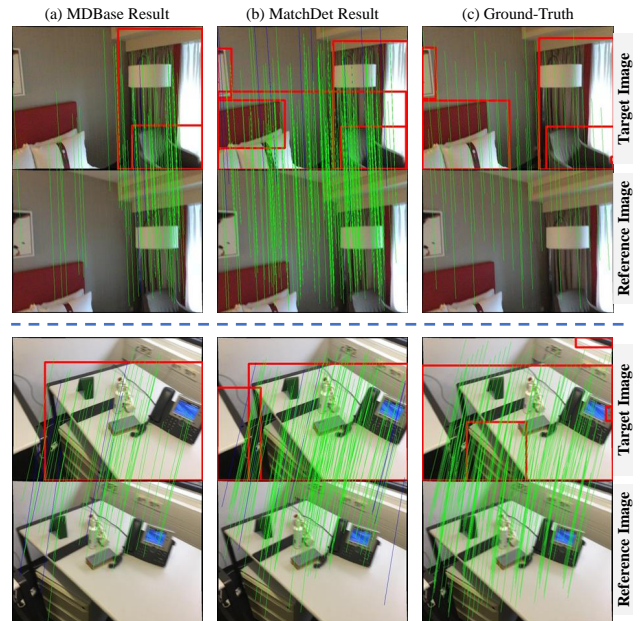


Figure 6: The visualization comparisons between our MatchDet and MDBase. Our MatchDet obtains better results of image matching and object detection.

## Visualization Analysis

Fig.6 presents the visualization comparisons between our MatchDet and MDBase, which shows that our MatchDet obtains better results of image matching and object detection. Fig.7 shows the visualizations for WAM, WSAM, Box Filter and MatchDet results.

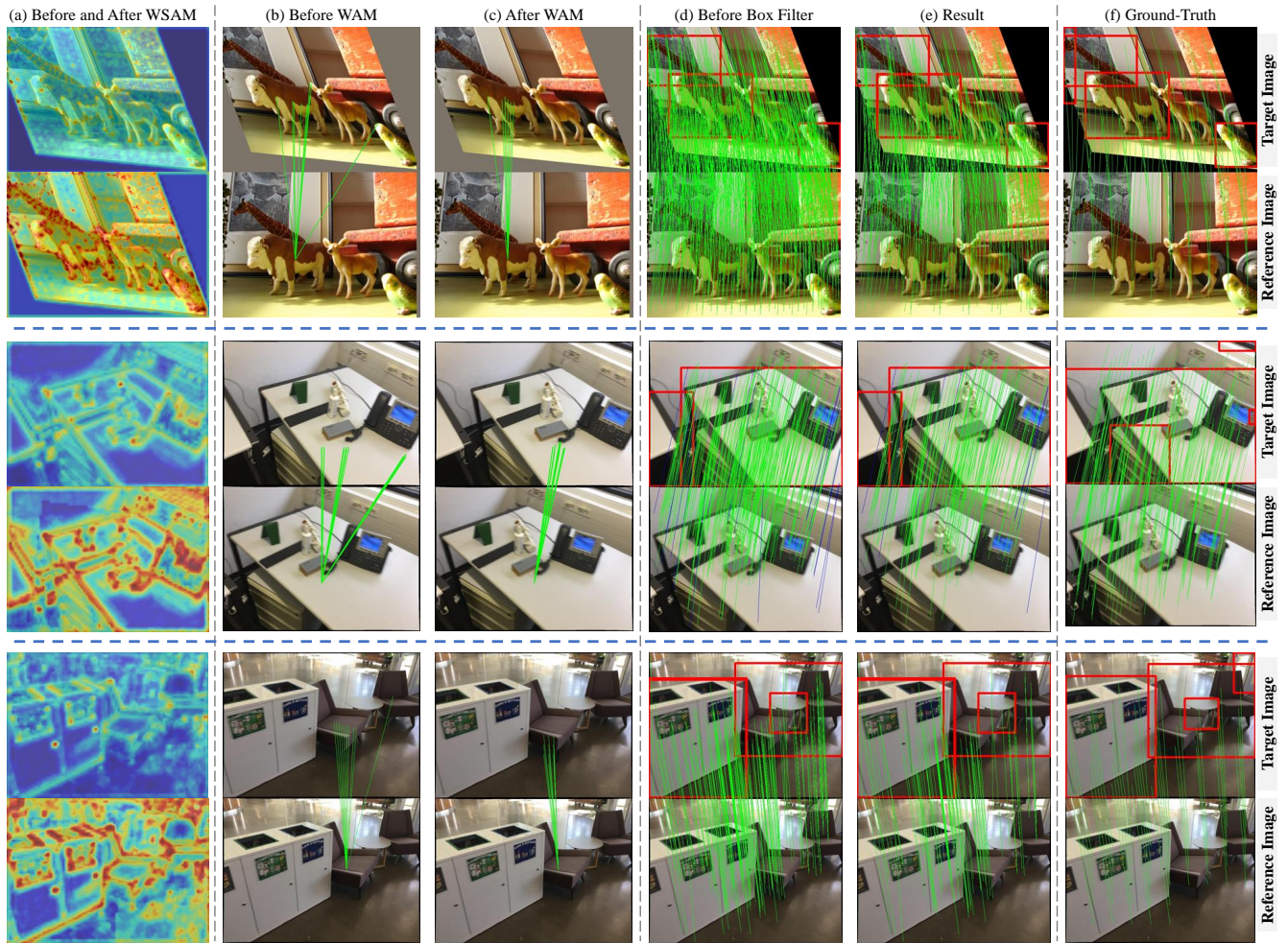


Figure 7: The visualizations for WAM, WSAM, Box Filter and MatchDet results under **GTBoxR** setting. (a) - (e), are the results processed by the corresponding modules before and after, respectively. (e) shows the predicted bounding boxes and matching results of MatchDet, where these matches are obtained after Box Filter. (f) is Ground-Truth. The blue color lines indicate wrong matches.