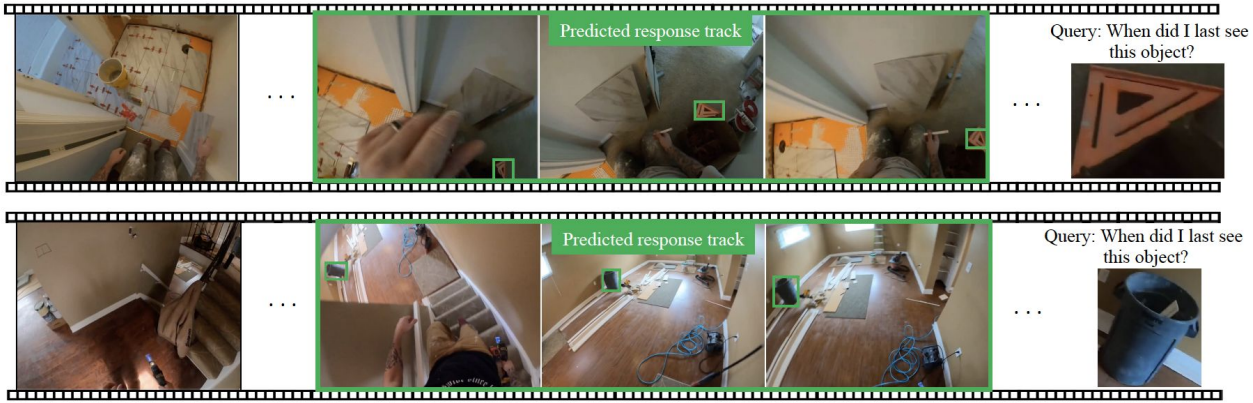
DLCV Final Project Research VQ2D From Ego4D Challenge

Team 4 :Baseliner

VQ2D

- Task
  - Given a **short clip** and an **image query**, predict the last time the object appeared in the clip
  - Predict the bounding box for each frame



VQ2D

Query: When did I last see this object?

image query

- Definitions
  - **clip**: a short video
  - **image query**: an image cropped from the clip, defined by **visual_crop**
  - **response_track**: a consecutive window when the object last appeared; the ground truth
  - **query_frame**: **the timestep** when I ask “When did I last see object X?”
- Task description
  - Your task is to predict the **response_track**
  - Find the specific object in the image query
    - If there are multiple trash cans in the video, only the one in the image query counts


```
{
  "0e7fba95-22d9-4ab0-9815-4bb7880d8557": {
    "annotations": [
      {
        "query_sets": {
          "1": {
            "is_valid": true,
            "query_frame": 91,
            "response_track": [
                {"frame_number": 53, "x": 560.46, "y": 127.76, "w
                {"frame_number": 54, "x": 535.53, "y": 38.24, "wi
                {"frame_number": 55, "x": 499.17, "y": 0.02, "wid
                {"frame_number": 56, "x": 445.86, "y": 0.04, "wid
                {"frame_number": 57, "x": 378.45, "y": 0.05, "wid
                {"frame_number": 58, "x": 426.04, "y": -0.01, "wi
            ],
            "object_title": "remote control",
            "visual_crop": {
                "frame_number": 126, "x": 411.31, "y": 254.65, "w
            }
```

Dataset

- TA Intro
- Submission
- Ego4D Official Page
- VQ2D Official Page
- [21/10] Ego4D
- [New] Query Image

Papers

- [18/07] CBAM
- [21/03] Spatio-Temporal Transformer
- [22/02] ActionFormer
- [22/08] Negative Frames Matters
- [22/11] Where is my wallet? (CoCoFormer)
- [22/12] EgoLoc (VQ3D)
- [23/05] Bayesian Decision Making (Hakuna Matata)
- [23/06] VQLoC (SOTA)

Code

- Official Code Guide
- Github Repo
- PaperWithCode (Not included VQ2D)
- VQLoC

Small Object Detection

- PaperWithCode
- [19/02] Copy-Paste
- [21/10] NWD
  - Code & 介紹
  - 介紹
  - Code 2

CoCoFormer

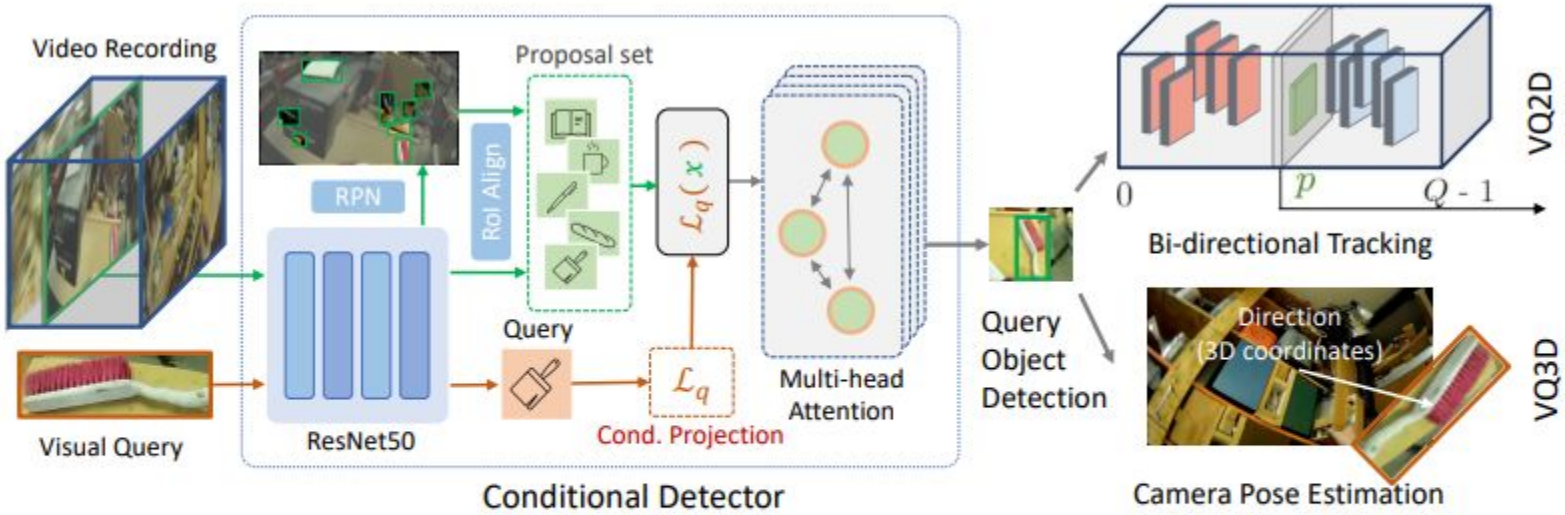


Figure 2. **Our detection + localization pipeline to solve the VQ task.** We firstly **detect** the query object in the video recording conditioned on the visual query, then apply bi-directional tracking from the detection result to **localize** the object in the video (VQ2D), or use camera pose estimation to **localize** the predicted bounding box in real-world coordinates (VQ3D). Specifically, our proposed CocoFormeris trained with our augmented query-frame pairs, and tested on all the frames of video. It has a conditional projection layer that generates unique transformation from the query, and a multi-head attention block on the query-based proposal embedding to exploit global context.

VQLoC

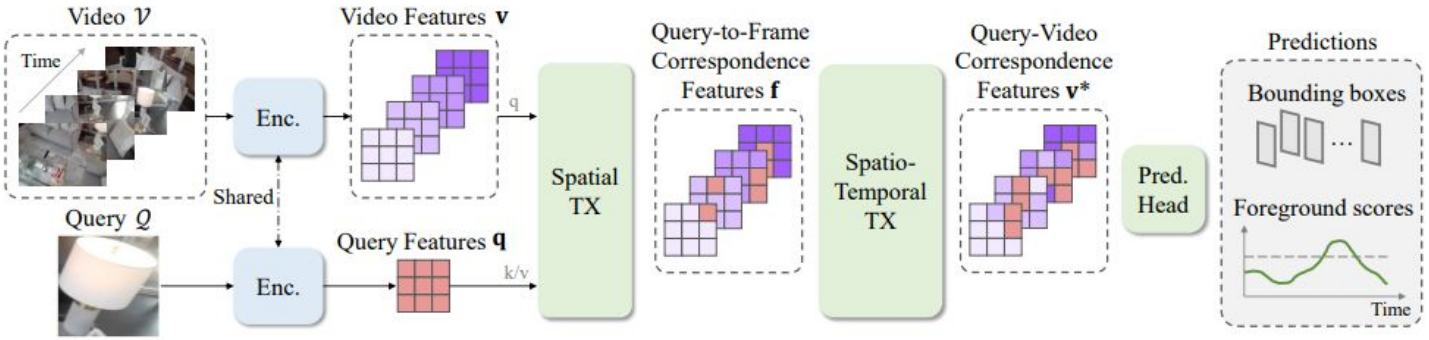


Figure 2: **Visual Query Localization from Correspondence (VQLoC)**: We address the VQL problem by first establishing the overall query-video relationship and then inferring the response to the query. VQLoC extracts image features for both query and video frames independently. It establishes query-to-frame correspondence between the query features $\mathbf{q}$ and each video frame feature $\mathbf{v}_i$ using a transformer module (Spatial TX), and obtains the correspondence features $\mathbf{f}$ for identifying potential query-relevant regions in the frames. VQLoC then propagates these frame-level correspondences through time using another transformer module (Spatio-Temporal TX), which leverages the frame-to-frame correspondence between nearby video frames and obtains the query-video correspondence features $\mathbf{v}^*$. Finally, VQLoC uses the query-video correspondence to predict the per-frame bounding box and the probability that the query object occurs in the frame.

Loss (Section 3.3)

Total Loss Limg =

(1) $\Sigma \hat{} b \in B \hat{} p [||b\hat{} c - bc|| + ||b\hat{} h - bh|| + ||b\hat{} w - bw|| + \lambda giou \cdot Lgiou(b\hat{}, b)] +$

(2) $\lambda p \cdot Lbce(P\hat{},P)$

where

(1) For all anchor box >thre_iou, calculate L1 loss and giou loss of bbox

(2) BCE loss for confidence score (both pos & neg, use hard negative mining)

The best_iou and best_prob_accuracy in validation() using pretrained model should be around 0.72 / 0.86 respectively.

Dataloader Shape (Sample in train_loader)

clip ===> torch.Size([3, 30, 3, 448, 448])

clip_with_bbox ===> torch.Size([3, 30])

before_query ===> torch.Size([3, 30])

clip_bbox ===> torch.Size([3, 30, 4])

query ===> torch.Size([3, 3, 448, 448])

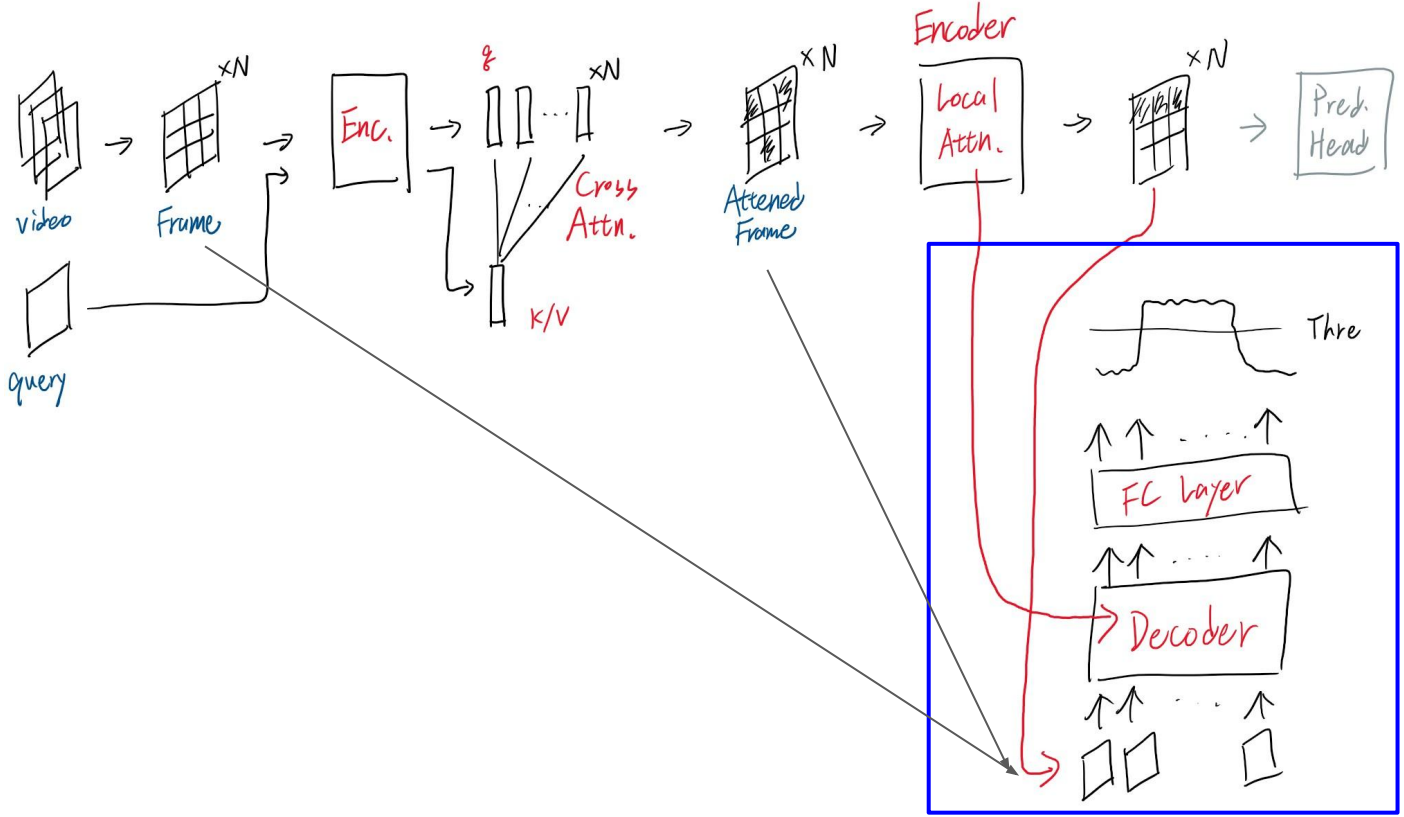
clip_h ===> torch.Size([3])

clip_w ===> torch.Size([3])

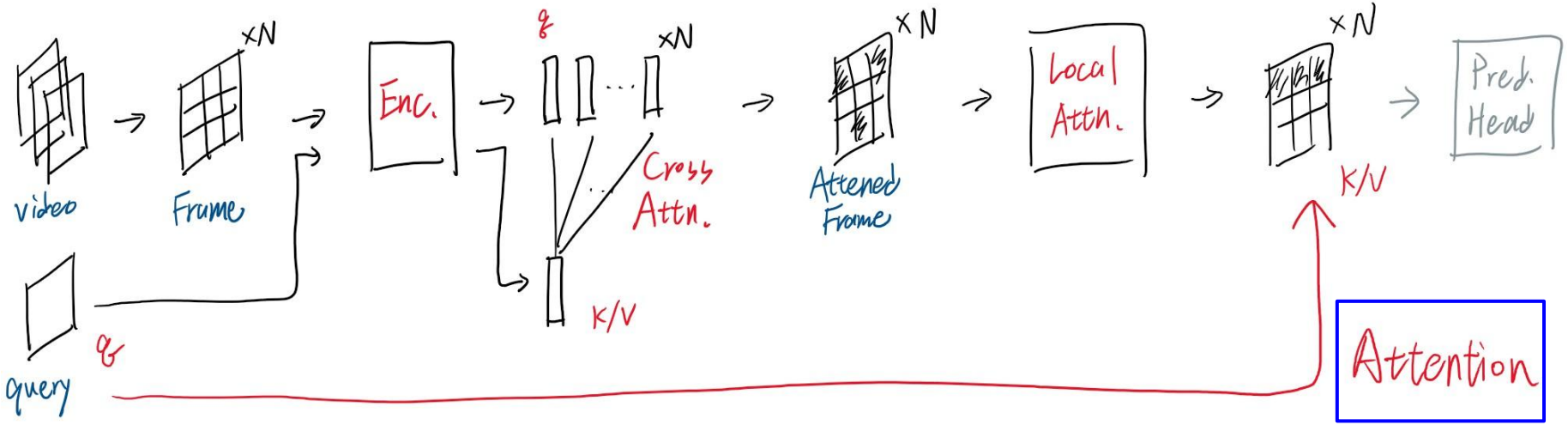
query_frame ===> torch.Size([3, 3, 448, 448])

query_frame_bbox ===> torch.Size([3, 4])

Proposed Method (1) - Add a Decoder Module



Proposed Method (2) - Add Attention Block



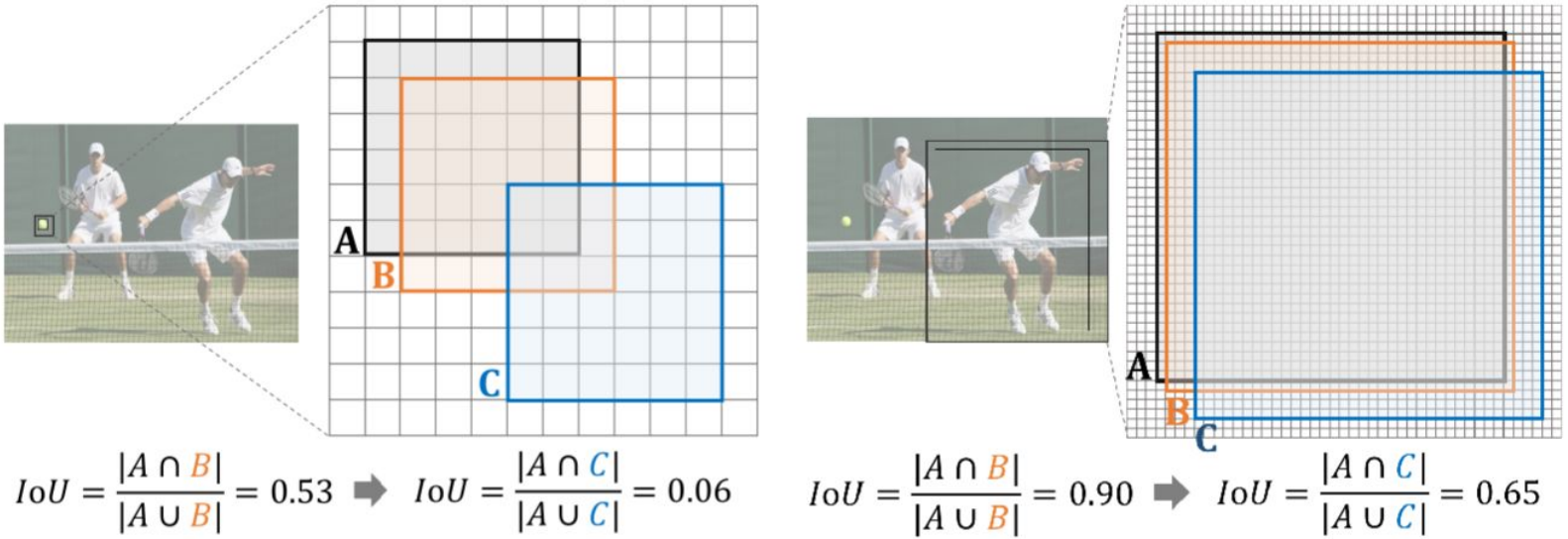
Proposed Method (3) - Small Object Enhancement

As shown in Table 7, CocoFormer achieves better performance in the case that query objects are small in the videos. In contrast, VQLoC demonstrates better accuracy on medium and large-size of query objects.

Table 7: Model performance on different scales of objects in the videos. *s*, *m* and *l* stands for small, medium and large object, respectively.

Method	Scale	tAP_{25}	$stAP_{25}$	rec%	Succ.
VQLoC	*s*	0.047	0.001	13.043	2.447
CocoFormer [50]	*s*	**0.067**	**0.030**	**19.565**	**21.113**
VQLoC	*m*	**0.213**	**0.138**	**44.719**	**33.738**
CocoFormer [50]	*m*	0.206	0.127	40.804	32.583
VQLoC	*l*	**0.454**	**0.387**	**67.680**	**53.635**
CocoFormer [50]	*l*	0.338	0.271	56.164	40.737

Proposed Method (3) - Small Object Enhancement (NWD)



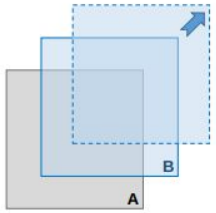
(a) Tiny scale object (b) Normal scale object

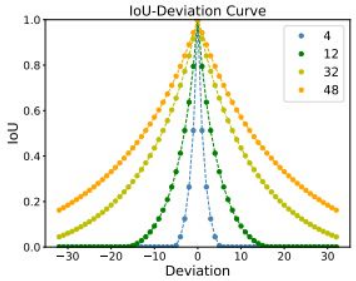
Figure 1: The sensitivity analysis of IoU on tiny and normal scale objects. Note that each grid denotes a pixel, box A denotes the ground truth bounding box, box B, C denote the predicted bounding box with 1 pixel and 4 pixels diagonal deviation respectively.

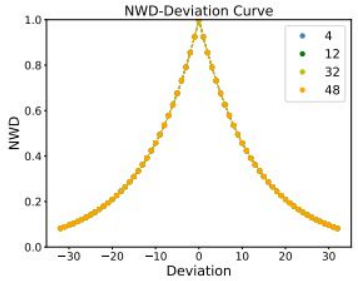
Proposed Method (3) - Small Object Enhancement (NWD)

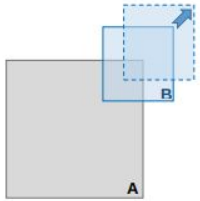
$$W_2^2(\mathcal{N}_a, \mathcal{N}_b) = \left\| \left(\left[cx_a, cy_a, \frac{w_a}{2}, \frac{h_a}{2} \right]^{\mathrm{T}}, \left[cx_b, cy_b, \frac{w_b}{2} \cdot \frac{h_b}{2} \right]^{\mathrm{T}} \right) \right\|_2^2.$$

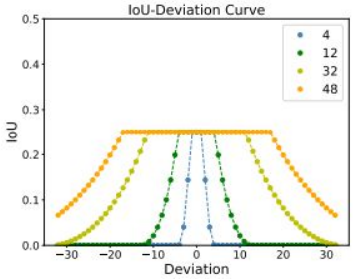
$$NWD\left(\mathcal{N}_a, \mathcal{N}_b\right) = \exp\left(-\frac{\sqrt{W_2^2\left(\mathcal{N}_a, \mathcal{N}_b\right)}}{C} \right)$$

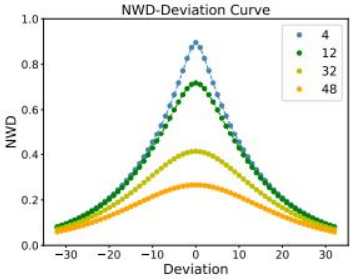










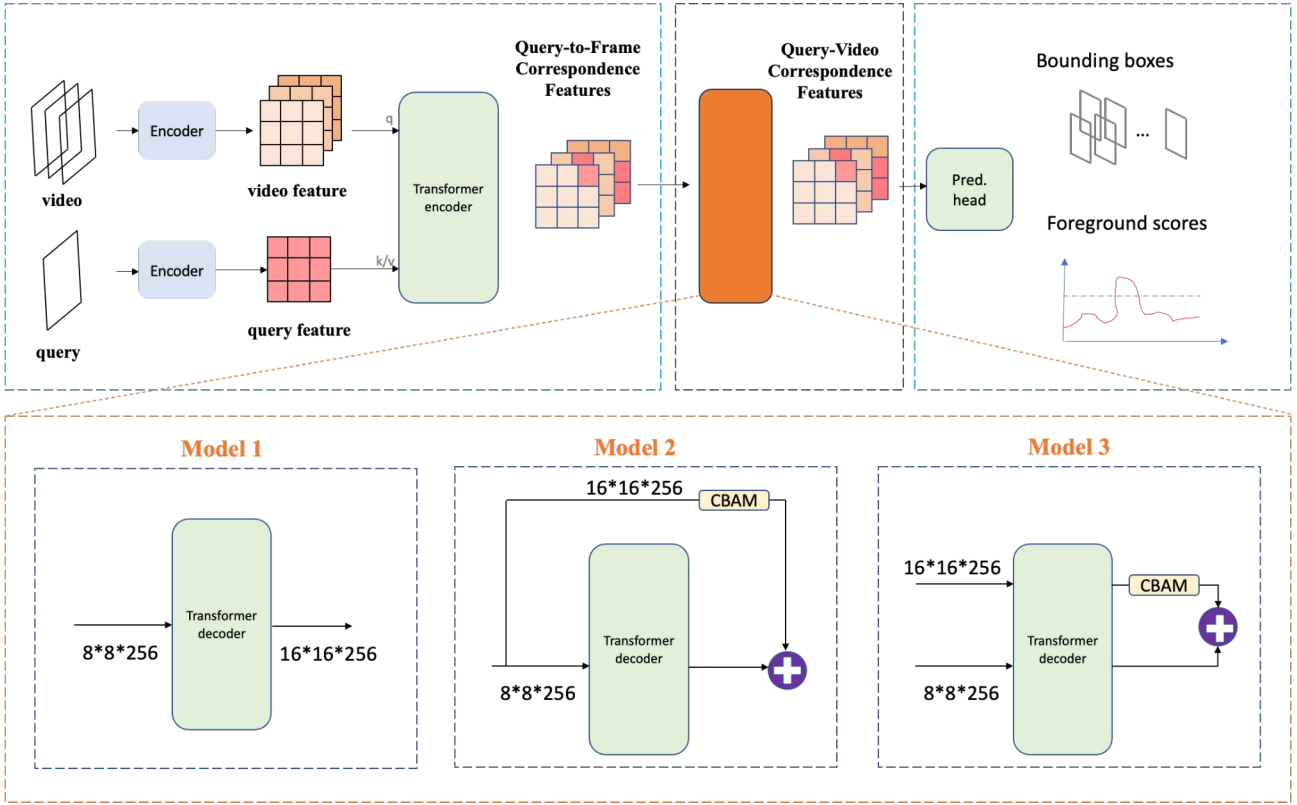


Proposed Method (4) - Regressor Head

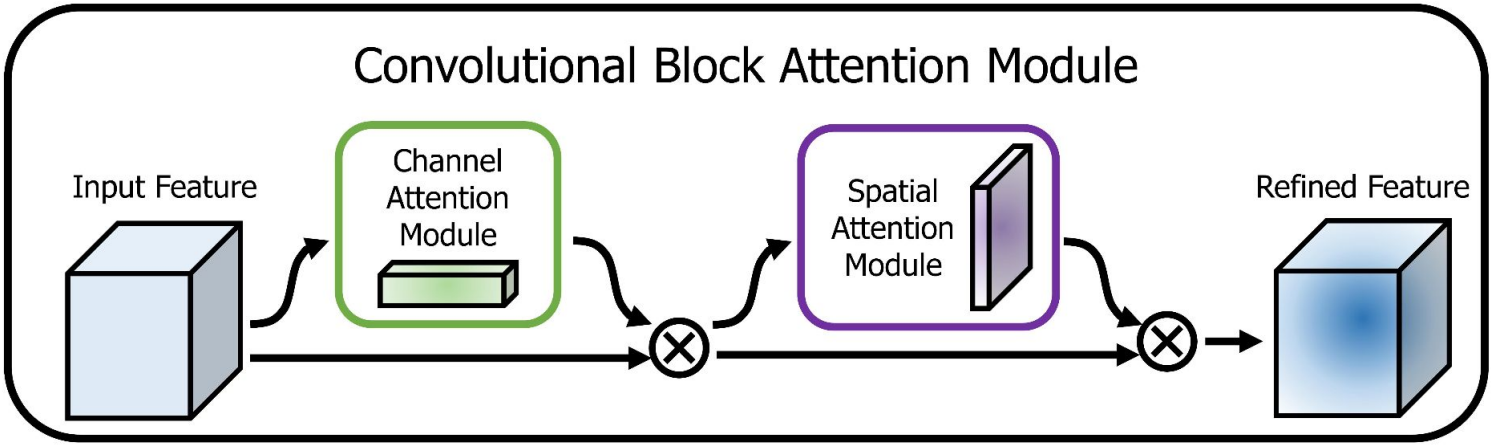
Add a regression head after attended frames, directly predict start frame and end frame.

Modify existing loss function by adding a L1 loss term.

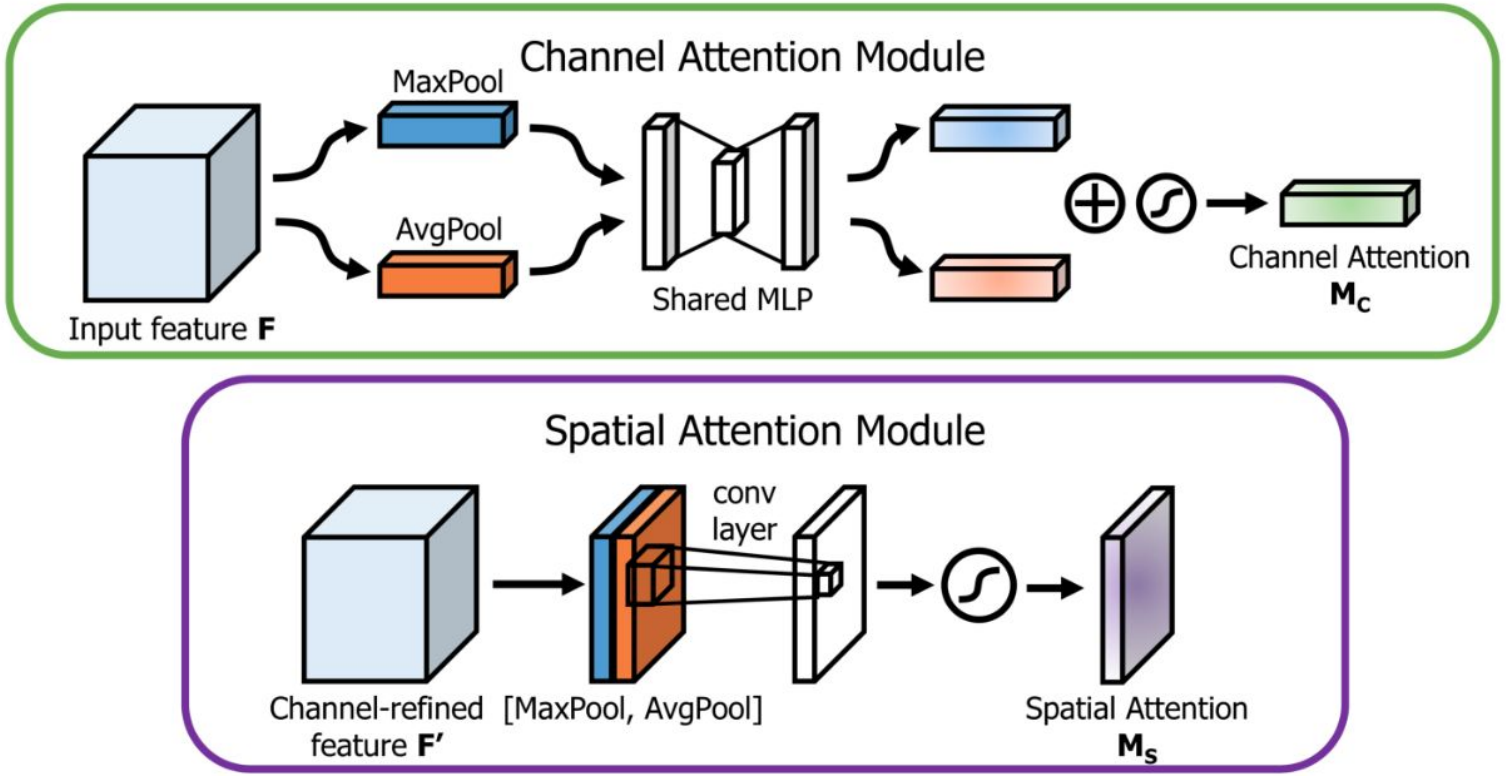
Proposed Method (5) - Model Structure for Small Object



Proposed Method (5) - Model Structure for Small Object



Proposed Method (5) - Model Structure for Small Object



Proposed Method (6) - DIoU loss

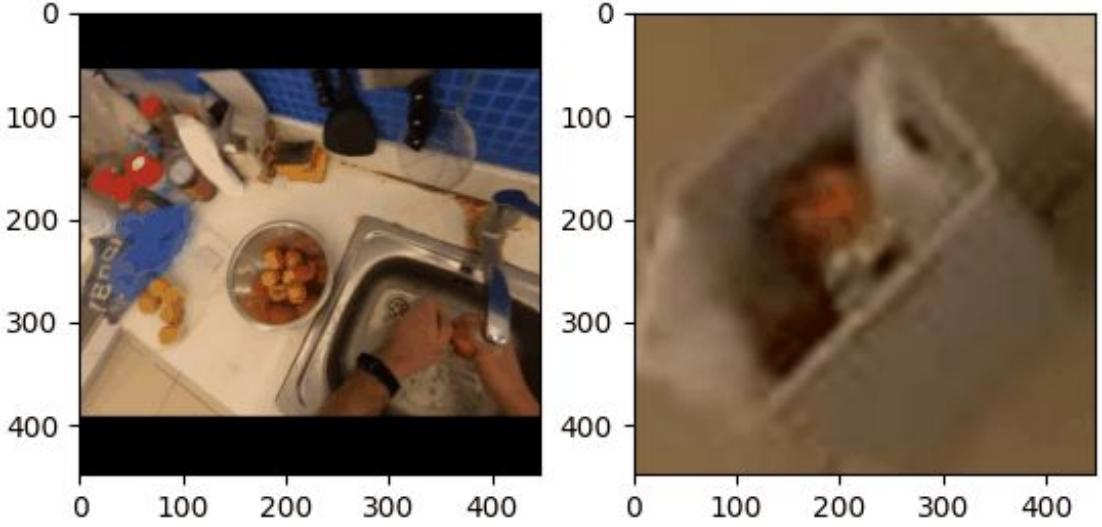
DIoU introduces a distance metric that takes into account the distance between the predicted box and the actual target box.

$$DIoU = IoU - \frac{\rho^2(b, b^{gt})}{c^2}$$

Result Comparison

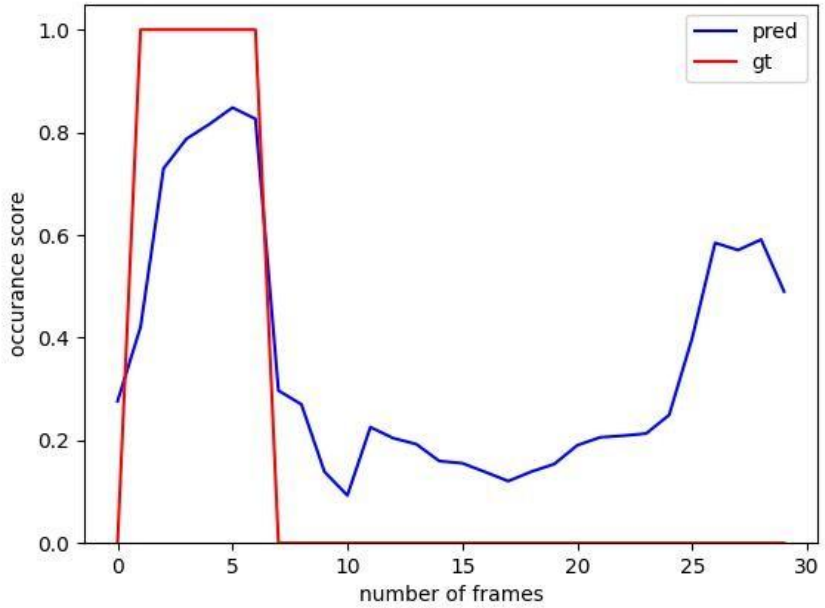
Method	epoch	Model	Valid iou	Valid prob	Test stAP
pretrain w/o finetune	-	prob	0.7213	0.8611	0.3502
pretrain w/ finetune	30	prob	0.6984	0.9278	0.2739
pretrain w/ attention layer	17	prob	0.4755	0.7972	0.1129
pretrain w/ NWD loss weight 0.3	6	prob	0.7350	0.9250	0.2986
pretrain w/ NWD loss weight 0.3	19	iou	0.7794	0.9194	0.3453
pretrain w/ NWD loss weight 10	2	prob	0.684	0.9167	0.2992
pretrain w/ NWD loss weight 10	6	iou	0.7705	0.8694	0.3345

Result Visualization

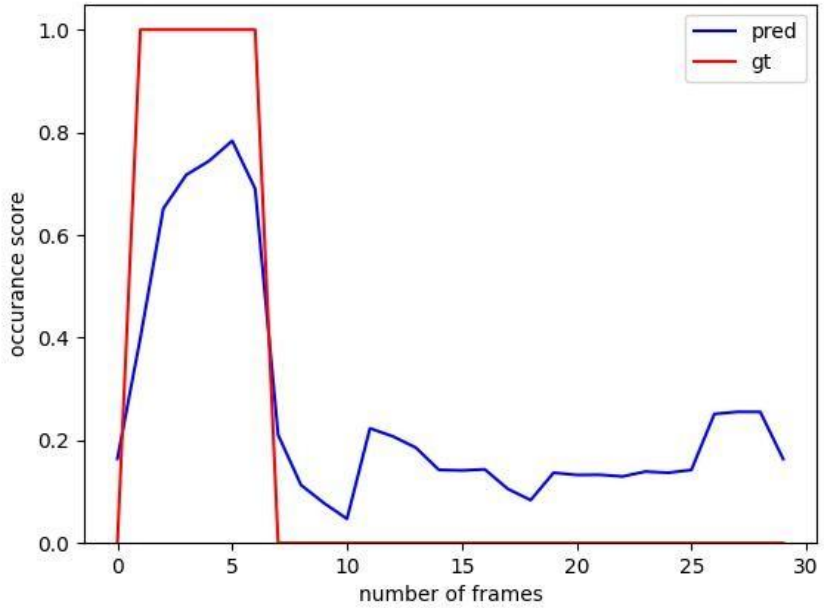


Comparison

Pretrained



NWD Loss



Comparison

From the previous figures, we can find that there are two peaks in the pretrained model prediction while there are only one in the NWD model. After examine the video and query image, we confirm that the second peak in the pretrained model prediction was a false positive (The query object does not appear in the video at all), thus we conclude that adding NWD in loss function help alleviate the problem of small object detection.