IDENTIFYING CONCEALED OBJECTS FROM VIDEOS

Anonymous authors
Paper under double-blind review

ABSTRACT

Concealed objects are often hard to identify from still images, as often camouflaged objects exhibit patterns seamless to the background. In this work, we propose a novel video concealed object detection (VCOD) framework, called **SLT-Net**, as the concealed state is likely to break when the object moves. The proposed SLT-Net leverages on both short-term dynamics and long-term temporal consistency to detect concealed objects in continuous video frames. Unlike previous methods that often utilize homography or optical flows to explicitly represent motions, we build a dense correlation volume to implicitly capture motions between neighbouring frames. To enforce the temporal consistency within a video sequence, we utilize a spatial-temporal transformer to jointly refine the short-term predictions. Extensive experiments on existing image and VCOD benchmarks demonstrate the architectural effectiveness of our approach. We further collect a large-scale VCOD dataset named **MoCA-Mask** with pixel-level handcrafted ground-truth masks and construct a comprehensive **VCOD benchmark** with previous methods. Videos and codes can be found at: [Link].

1 INTRODUCTION

Video Concealed Object Detection (VCOD) is the task of discovering objects in a video that appearance-wise exhibit a great deal of similarity to the background scene. Despite enjoying wide applications (*e.g.,* surveillance and security (Liu et al., 2019a) , autonomous driving (Ranjan et al., 2019), medical image segmentation (Fan et al., 2020b; Wu et al., 2021), locust detection (Kumar & Rahman, 2021) and robotics (Michels et al., 2005)), the problem of COD is a daunting task as concealed objects are often indistinguishable to naked-eyes. This, in turn, has made VCOD a relatively new and unexplored problem in computer vision, as compared to several related problems such as video object detection (VOD) (Yang et al., 2019; Beery et al., 2020), video salient object detection (VSOD) (Ji et al., 2021b) and video motion segmentation (VMS) (Yang et al., 2021) tasks.

In majority of problems in computer vision (*e.g.,* instance segmentation, saliency detection), it is assumed that objects have clear boundaries. This allows us to formulate the problem at the image-level, and even consider improvements if motion information is available to us. In contrast, object boundaries are ambiguous and indistinguishable when it comes to detecting concealed objects. This not only makes detection from images challenging, but also results in inaccurate estimation of optical flow and motion information in videos (Lamdouar et al., 2020).

The lack of clear boundaries means that the appearance of the concealed object resembles the background. This shows itself as two fundamental difficulties: **1)** the object boundaries are often seamlessly blended into the background and is observable only when the object moves; **2)** the object often has repetitive textures to the environment, hence determining the movement of pixels across frames to estimate the motion (*e.g.,* as done in optical flow) is erratic and erroneous. Therefore and to successfully address VCOD, a neural network needs to successfully discovers the nuances between the concealed object and the background with the help of motion information as a result of the first difficulty. However, the motion information is itself noisy and inaccurate according to the second difficulty. As such, employing techniques developed for VOD, VSOD, and VMS may fail miserably if naively employed or combined to address the problem VCOD.

In this work, we introduce a novel VCOD framework (**SLT-Net**) that utilize a short-term dynamics and long-term temporal consistency to detect concealed objects in videos. Specifically, we employ a short-term dynamic module to *implicitly* capture the motion between consecutive frames. Rather than using optical flow to *explicitly* represent motions, we use a full-range correlation pyramid strat-

egy to implicitly represent them. The main motivation behind the use of a correlation-pyramid is that even SOTA optical flow algorithms fail to estimate motions for concealed objects and their errors get accumulated over the duration of the video. To provide stable estimation, we further introduce a long-term refinement module to alleviate the effect of accumulated inaccuracies that occurred in the short-term dynamic module.

We realize the SLT-Net as a hybrid neural network with both transformer and CNN components. In particular, we use a transformer structure to encode features for constructing correlation pyramid. Aside from its design flexibility, features extracted by the transformer contain global contextual information with long-range dependencies and less inductive bias (Wang et al., 2021b), which we observe to be more distinguishable in estimating the motion.

While the correlation pyramid strategy can effectively capture motions for detecting concealed objects, it cannot scale gracefully to long video sequences due to its computational complexity. To solve this issue, we adopt a sequence-to-sequence model with a spatial-temporal transformer to refine the pair-wise prediction with long-term consistency across the videos as we empirically find it is more accurate than standard ConvLSTM model.

Being an emerging problem, no large-scale dataset was available to evaluate and benchmark VCOD systems. To promote new developments in this domain, we have curated a large-scale VCOD dataset based on the Moving Camouflaged Animals (MoCA) (Lamdouar et al., 2020). The new dataset, or **MoCA-Mask** for short, contains 87 video sequences with 22,939 frames in total with pixel-level GT masks. MoCA-mask encapsulates a variety of challenges such as complex background, and tiny and well camouflaged objects. We provide annotations, bounding boxes and dense segmentation masks for every 5^{th} frame for all the videos in the dataset. For the frames without dense annotations, we provide pseudo GT masks generated by a SOTA optical flow algorithm (Teed & Deng, 2020). We also provide the first comprehensive benchmark for existing VCOD methods.

In a nut, our contributions are as follows:

- A new framework called **SLT-Net** and achieve the new SOTA performance.
- The largest-scale **MoCA-Mask** dataset for the challenging VCOD task.
- The comprehensive **VCOD benchmarks**, which can facilitate the progress of this field.

2 Related Works

COD. As the opposite of "salient" object detection, "concealed" object detection aims to identifying a camouflaged object from its background. Prior work ANet (Le et al., 2019) incorporated classification information into representation learning. MGL (Zhai et al., 2021) presented a joint learning framework for detecting camouflaged objects and their edges by two mutual graph-based modules. PFNet (Mei et al., 2021) introduced a novel mining strategy to discover false-positive predictions and remove false-negative ones. SINet (Fan et al., 2020a) address the problem by first performing a coarse search for camouflaged objects and then refining it by segmentation. An improvement called SINet-v2 (Fan et al., 2021a) was proposed with the neighbor connection decoder and the reverse attention mechanism. Lately, Lv et al. (2021) introduced two new tasks for camouflaged object detection, namely camouflaged object discriminative region localization and camouflaged object ranking, along with relabeled new NC4K testing dataset.

SOD. In order to find the most visually distinctive objects in an image, classical studied used hand-crafted features. A deep network is often trained to benefit the task in one of the following aspects: **1)** to learn better salient object edges or features, *e.g.,* EGNet (Zhao et al., 2019), BASNet (Qin et al., 2019) **2)** to learn better refinement network, *e.g.,* RFCN (Wang et al., 2016), CPD (Wu et al., 2019) **3)** to better handle scale variation, *e.g.,* DSS (Hou et al., 2017), GateNet (Zhao et al., 2020) **4)** to better integrate the information, *e.g.,* DCL (Li & Yu, 2016), TSPOANet (Liu et al., 2019b), ICON (Zhuge et al., 2021).

VSOD. To detect salient objects in videos, DLVS (Wang et al., 2017) introduced fully convolutional networks for pixel-wise saliency prediction. DSR3 (Le & Sugimoto, 2017) exploited an end-to-end 3D neural network to produced video sequences, which incorporates 3D CNN modules combined with recurrent refinement units to predict saliency maps. To better learn temporal information over frames, following works considered SpatioTemporal CRF (Le & Sugimoto, 2018), pyramid dilated convLSTM (Song et al., 2018) in the design of their networks. FGRN (Li et al., 2018), RCRNet (Yan

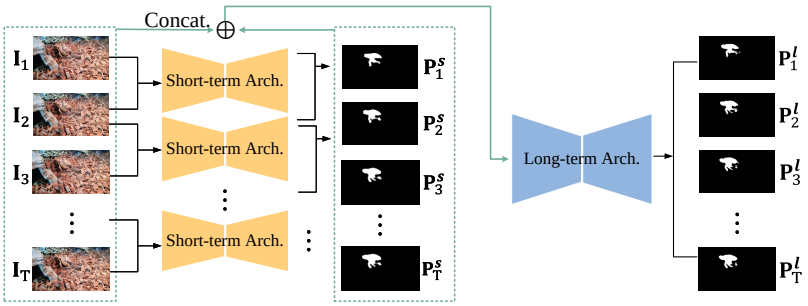


Figure 1: The overall pipeline of the SLT-Net. The SLT-Net consists of a short-term detection module and a long-term refinement module. The short-term detection module takes a pair of consecutive frames and predicts the concealed object mask for the reference frame. The long-term refinement module takes T predictions from the short-term detection module along with their corresponding referenced frames to generate the final predictions.

et al., 2019) adopted extra flow guided networks to improve temporal coherence. Later, SSAV (Fan et al., 2019) specifically focused on saliency shift phenomenon and established a comprehensive benchmark for VSOD. FSNet (Ji et al., 2021b) leveraged the mutual constraints of appearance and motion cues, demonstrating superior performances to many existing methods.

VMS. The task of VMS focuses on discovering moving objects in dynamic videos. Traditional approaches normally address this problem by extracting motion boundaries in the flow field and then refining the initial estimate with appearance features (Papazoglou & Ferrari, 2013), or combining motion and appearance information by a fusion architecture (Jain et al., 2017). Another line of work explicitly leverage optical flow as the input to train a CNN based network and generate pixel-level motion labels rely on supervised learning, *i.e.,* (Tokmakov et al., 2017) or in an unsupervised manner, *i.e.,* (Yang et al., 2021).

VCOD. Different from VMS, visual cues of camouflage tasks are considered less effective than motion cues. Prior works mainly relied on homography or optical flows to detect motion patterns. (Bideau & Learned-Miller, 2016; Bideau et al., 2018) segment moving objects from environment by approximating different motion models computing from dense optical flow. In particular, (Bideau & Learned-Miller, 2016) proposed a two step segmentation algorithm, which first compensated for the camera rotation and then segmented the angle of the optical flow into objects and the background. Although each motion model is updated with orientations of optical flow over the time, the initial motion is heuristic. In (Bideau et al., 2018), a trainable network to segment from the angle field rather than raw optical flow is developed. (Lamdouar et al., 2020) proposed a video registration and motion segmentation framework, along with a larger camouflaged dataset (MoCA) labelled by bounding boxes for every 5^{th} frame. The explicit alignment method by optical flow builds spatial correspondence between neighboring frames. However, the optical flow estimation may not be accurate enough to support effective alignment, particularly in dynamic scenes with fast object motions.

3 The Proposed Framework

In this section, we provide a detailed description of our framework. The input of our SLT-Net is a video clip that contains concealed objects, and the output is a set of pixel-wise binary masks of the concealed objects for each frame in the video. Specifically, let us denote the video clip with T frames by $\{\mathbf{I}^{\mathrm{t}}\}_{t=1}^{\mathrm{T}}, \mathbf{I}^{\mathrm{t}} \in \mathbb{R}^{3\times H\times W}$, where H, W are the height and the width of the frame. Our network is to assign a binary mask $\mathbf{M}^{\mathrm{t}} \in \{0,1\}^{H\times W}$ for the video frame $\mathbf{I}^{\mathrm{t}}$ at time t.

3.1 Overview

The overall framework of the SLT-Net is shown in Figure 1. The SLT-Net consists of a short-term detection module and a long-term refinement module. The short-term detection module takes a pair of consecutive frames and predicts the concealed object mask for the reference frame[1]. Then the

[1]The result for the last frame can be achieved by swapping the reference frame for the last pair.

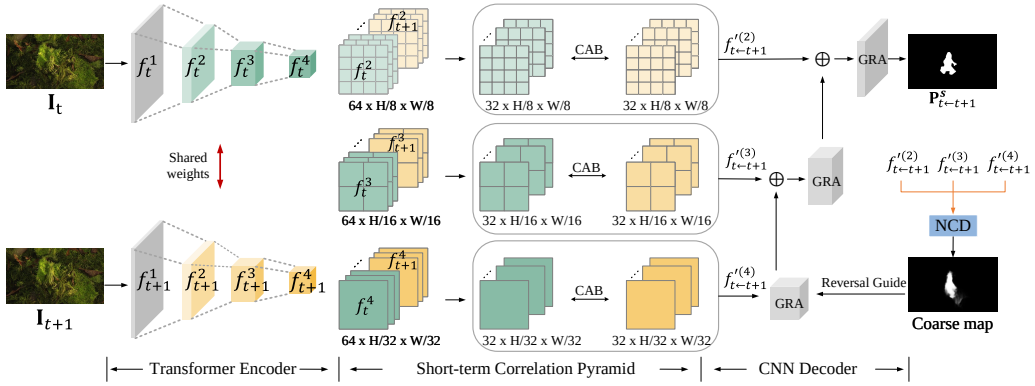


Figure 2: The overview of our short-term network pipeline. The network first extracts features from the input frames by a transformer encoder, then computes a full-range volumetric correspondence between the reference frame $\mathbf{I}_t$ and its neighboring frame $\mathbf{I}_{t+1}$ to form a correlation volume pyramid. A CNN decoder is used to predict the final prediction from the motions captured by the short-term correlation pyramid.

long-term refinement module takes T predictions from the short-term detection module as well as their corresponding referenced frames to generate the final prediction results. To train the SLT-Net, we adapt a two-stage strategy. We first train the short-term detection module using our pixel-wise annotations only. Once the network converges, we attach the long-term refinement module to the SLT-Net and train the whole model while fixing the short-term detection network.

3.2 Short-term Architecture

We illustrate our short-term architecture in Figure 2. It takes two consecutive frames as input from a video and predicts a binary mask of the reference frame. Our model consists of three main modules: (1) **Transformer Encoder** for feature extraction; (2) **Short-term Correlation Pyramid** for capturing short-term dynamics; and (3) **CNN Decoder** to predict the short-term segmentation. Below we describe details of each module.

(1) **Transformer Encoder.** We adapt a Siamese structure with the pyramid vision transformer (PVT) (Wang et al., 2021a) to extract features from two consecutive frames. The encoder consists of four stages that generate feature maps at four different scales. All stages share a similar structure, including a patch embedding layer and transformer blocks. The sizes of the features at each stage are $C_i \times H/2^{i+1} \times W/2^{i+1}$, $i \in \{1, 2, 3, 4\}$, where the H, W, C represent the height, the width and the channels. We set $C = 32$ in our experiments. Following (Fan et al., 2021a), we adapt three texture enhanced modules (TEM) for the features from the last three stages. To attain more discriminative feature representations, each TEM includes four parallel residual branches.

(2) **Short-term Correlation Pyramid.** Prior works, *i.e.,* (Tokmakov et al., 2017; Yang et al., 2021) explicitly incorporate motion by taking optical flow from consecutive frames as the inputs into a deep network. However, inaccurate optical flow may result in error accumulation at subsequent predictions. If we would like to jointly optimize the optical flow module with the segmentation module, the ground truth of optical flow will be needed. To solve this issue, we propose a correlation pyramid to implicitly capture motion information. Since the features that form the correlation pyramid will be updated with the segmentation ground truth. The motion estimation will be updated accordingly.

As shown in Figure 3, a correlation aggregation block (CAB) $\mathbf{C}$ is defined as the core unit of the short-term correlation pyramid. It allows us to find correspondences at a global scale. Given a pair of frame features$\{f_t, f_{t+1}\} \in \mathbb{R}^{C \times H' \times W'}$, the 4D correlation volume $\mathbf{C}(\mathbf{I}_t, \mathbf{I}_{t+1}) \in \mathbb{R}^{H' \times W' \times H' \times W'}$ is defined as:

$$\mathbf{C}(\mathbf{I}_t, \mathbf{I}_{t+1})_{xyuv} = \exp\left(\sum_c \mathcal{F}_\theta(\mathbf{I}_t)_{xyc} \cdot \mathcal{F}_\theta(\mathbf{I}_{t+1})_{uvc}\right), \tag{1}$$

with c being the index along the channel dimension of frame features. It pairs up all the pixels in the neighboring features and computes their correlation by a radial basis function kernel.

Next, we aggregate the features based on the correspondence, by normalizing the correlation volume $\mathbf{C}(\mathbf{I}_t, \mathbf{I}_{t+1})_{xyuv}$ along the last two dimensions uv over their sum. The normalized correlation

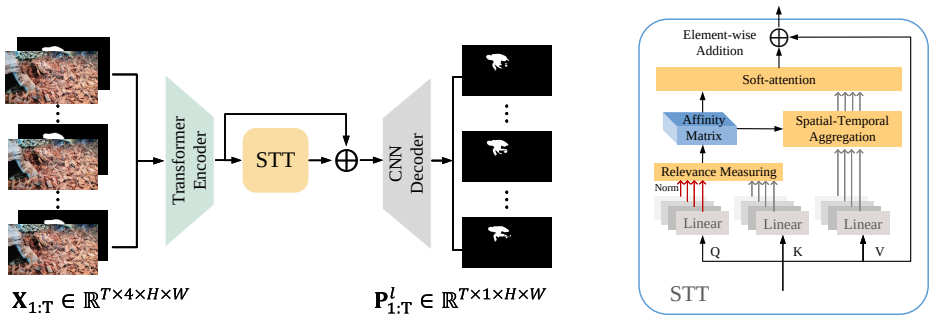


Figure 4: The overview of the proposed long-term consistency architecture. It formulates the process as a seq-to-seq problem and refines the pair-wise predictions with a spatial-temporal transformer.

volume is computed as follows:

$$\tilde{\mathbf{C}}(\mathbf{I}_t, \mathbf{I}_{t+1})_{xyuv} = \frac{\mathbf{C}(\mathbf{I}_t, \mathbf{I}_{t+1})_{xyuv}}{\sum_u \sum_v \mathbf{C}(\mathbf{I}_t, \mathbf{I}_{t+1})_{xyuv}}. \tag{2}$$

Figure 3 only shows a correlation on one scale. To make the network learn more detailed information, we construct a correlation pyramid $\{\mathbf{C}^i\}, i \in \{2, 3, 4\}$ by incorporating the extracted multi-scale features from the transformer encoder.

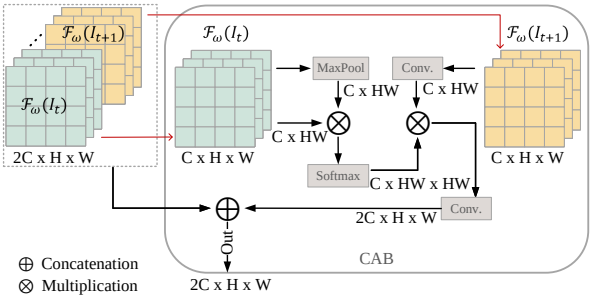


Figure 3: Correlation aggregation block (CAB) computes the normalized correlation volume of feature maps between the reference frame (green blocks) and the target frame (yellow blocks).

(3) **CNN Decoder.** As shown by (Fan et al., 2021a), the neighbor connection decoder is more reliable than conventional connection decoder (*i.e.,* densely connection or short connection). In addition, the group-reversal attention (GRA) strategy used in (Fan et al., 2021a) can provide more accurate segmentation results around the object boundaries. Based on these, we directly feed features from the short-term correlation pyramid, *i.e.,* $\{f'^{(i)}_{t\leftarrow t+1}\} \in \mathbb{R}^{C\times H/2^{i+1}\times W/2^{i+1}}, i \in \{2, 3, 4\}$, into the GRA blocks, and generate refined feature maps, progressively. The neighbor connection decoder (NCD) is used to generate a coarse map, which could provide reversal guidance of rough location of the concealed object. In this way, we gather the low-level features from the CNN decoder and the high-level features from the correlation pyramid.

Learning Strategy. We train the short-term training stage by minimizing the loss below:

$$\mathcal{L} = \mathcal{L}^w_{ce} + \mathcal{L}^w_{iou}. \tag{3}$$

The weighted intersection-over-union loss $\mathcal{L}^w_{ce}$ increases the weights of hard pixels to emphasize their importance. The weighted binary cross entropy loss $\mathcal{L}^w_{iou}$ pays more attention to hard (*i.e.,*, uncertainty) pixels rather than assigning all pixels with equal weights. Readers could refer to prior work (Wei et al., 2020) to find more details regarding the definitions of these two loss functions.

3.3 Long-term Consistency Architecture

To encourage long-term temporal consistency, we introduce a refinement network with the spatial-temporal information to generate final predictions. Given a sequence of $\mathbf{I}_{1:T} = \{\mathbf{I}_1, \mathbf{I}_2, \ldots, \mathbf{I}_T\}$ and the pixel-wise predictions of $\mathbf{P}^s_{1:T} = \{\mathbf{P}^s_1, \mathbf{P}^s_2, \ldots, \mathbf{P}^s_T\}$ from our short-term architecture, we formulate the long-term consistency refinement process as a seq-to-seq problem.

Figure 4 illustrates the long-term consistency architecture. We use the same backbone as the short-term architecture, *i.e.,* transformer encoder and CNN decoder modules, since it has been already pre-trained on concealed datasets that could largely accelerate the long-term training processing. For each frame of the input sequence, we concatenate the color frame $\mathbf{I}_t$ with its corresponding prediction $\mathbf{P}^s_t, t \in [1 : T]$ on the channel dimension, and then stack every concatenated frame within the sequence to form a 4D tensor $\mathbf{X}_{1:T} \in \mathbb{R}^{T\times 4\times H\times W}$. The network takes $\mathbf{X}_{1:T}$ as the input and output the final prediction sequence $\mathbf{P}^l_{1:T} \in \mathbb{R}^{T\times 1\times H\times W}$.

There are two kinds of seq-to-seq modeling architecture: one is to use convLSTM to model the temporal information, and the other is to use a transformer. We implement both architectures and compare their results in Section 4.3. We empirically find that using the transformer structure can lead to better results, so we select the spatial-temporal transformer (STT) (Ji et al., 2021a) to enforce the long-term consistency.

We show the details of STT on the right side of Figure 4. For each target pixel, to reduce the complexity for building a dense spatial-temporal affinity matrix, we select a fixed number of relevance measuring blocks to construct the affinity matrix within a constrained neighborhood of it.

We apply the hybrid loss (Fan et al., 2021b) during the training:

$$\mathcal{L}_{hybrid} = \mathcal{L}^w_{ce} + \mathcal{L}^w_{iou} + \mathcal{L}_e, \tag{4}$$

where $\mathcal{L}_e$ is the Enhanced-alignment loss. The hybrid loss can guide the network to learn pixel-, object- and image-level features.

3.4 Semi-supervised Training Procedure

As the annotations are provided in the form of dense segmentation masks every 5^{th} frame, we adopt a bi-directional consistency check strategy to generate pseudo masks for unlabelled frames. Given five consecutive frames $\{\mathbf{I}_t, \mathbf{I}_{t+1}, \mathbf{I}_{t+2}, \mathbf{I}_{t+3}, \mathbf{I}_{t+4}\}$ and labelled ground-truth $\mathbf{gt}_t$, we first estimate forward and backward optical flow fields between frame $\mathbf{I}_t$ and $\mathbf{I}_{t+n}, n \in [1, 4]$. Then we can produce the warped ground-truth $\hat{\mathbf{gt}}_{t+n}$ with the inverse warping from ground-truth $\mathbf{gt}_t$.

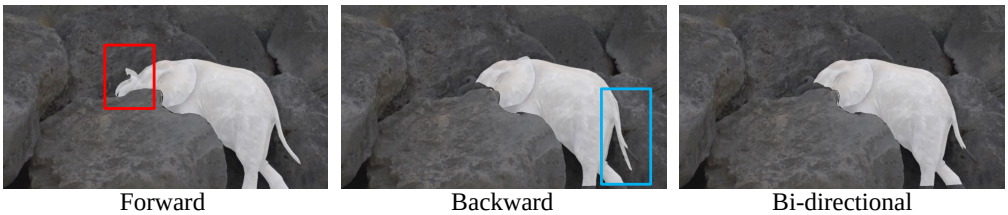
Step 1 - Flow Estimation. We take the ground-truth mask of the reference frame $\mathbf{I}_t$ as an example, to generate pseudo ground-truth of its immediate following frame $\mathbf{I}_{t+1}$. The optical flow estimation module[2] $\mathcal{O}$ takes $\mathbf{I}_t$ and $\mathbf{I}_{t+1}$ and predicts the optical flow field:

$$\mathbf{u}^x_{t,t+1},\ \mathbf{u}^y_{t,t+1} = \mathcal{O}(\mathbf{I}_t, \mathbf{I}_{t+1}), \tag{5}$$

where $\mathbf{u}^x_{t,t+1}$ and $\mathbf{u}^y_{t,t+1}$ denote the x, y components of the estimated flow field, respectively. The flow field maps each pixel $(x,\ y)$ in $\mathbf{I}_{t+1}$ to its corresponding coordinates $(x',\ y') = (x + \mathbf{u}^x_{t,t+1}(x),\ y + \mathbf{u}^y_{t,t+1}(y))$ in $\mathbf{I}_t$.

Step 2 - Forward / Backward Pseudo Ground-truth. Given the forward optical flow sequences $(\mathbf{flow}_t, \mathbf{flow}_{t+n}), n \in 1, 2, 3, 4$, we can obtain the aligned neighboring frame $\hat{\mathbf{gt}}_{t+n}$ by a warping interpolation on $\mathbf{gt}_t$ using the mapped coordinates. After repeating the explicit alignment step for the preceding frame, we acquire the sequence of warped input frames $\{\mathbf{gt}_t, \hat{\mathbf{gt}}_{t+1}, \hat{\mathbf{gt}}_{t+2}, \hat{\mathbf{gt}}_{t+3}, \hat{\mathbf{gt}}_{t+4}\}$. The backward pseudo ground-truth sequences are obtained by performing warping ground-truth masks with backward optical flows in the reverse order.

Step 3 - Bi-directional Consistency Check. To identify valid masks, we adopt forward-backward consistency check to eliminate inconsistent regions. Under the forward-backward consistency assumption (Sundaram et al., 2010), traversing flow vector forward and then backward should arrive at the same position. We mark pixels as invalid whenever this constraint is violated. As shown in Figure 5, the invalid regions emphasized by the orange boxes are marked as background.



Forward Backward Bi-directional

Figure 5: Illustration of forward-backward consistency check. After bi-directional check, undesirable ghosting artifacts, *i.e.,* the nose (red box) of the elephant in forward direction and the tail (blue box) in backward direction, and occlusions can be effectively removed.

[2]In practice, we make use of RAFT (Teed & Deng, 2020) to obtain the optical flow.

4 EXPERIMENTS

Metrics. We adopt following evaluation metrics to measure the pixel-wise masks, including: (1) MAE (M), which assesses the pixel-level accuracy between prediction and labeled masks. (2) Enhanced-alignment measure (E_ϕ) (Fan et al., 2018) , which simultaneously evaluates the pixel-level matching and image-level statistics. This metric is naturally suited for assessing the overall and localized accuracy of the concealed object detection results. Note that we report mean E_ϕ in the experiments. (3) S-measure (S_α) (Fan et al., 2017), which evaluates region-aware and object-aware structural similarity. (4) Weighted Fmeasure F_β^w (Margolin et al., 2014) can provide more reliable evaluation results than the traditional F_β. (5) meanDice, which measures the similarity between two sets of data. (6) meanIoU, which measures the overlap between two masks.

Baseline Models. We select nine cutting-edge baselines, including **I.** six image based methods *i.e.,* EGNet (Zhao et al., 2019), BASNet (Qin et al., 2019) , CPD (Wu et al., 2019), PraNet (Fan et al., 2020b), SINet (Fan et al., 2020a), SINet-v2 (Fan et al., 2021a), and **II.** three video based methods, *i.e.,* PNS-Net (Ji et al., 2021a), RCRNet (Yan et al., 2019), and MotionGroup (Yang et al., 2021). Please refer to the ***Appendix*** for our implement details.

4.1 DATASETS

CAD. Camouflaged Animal Dataset (CAD) is a small set of camouflaged animals, first introduced by (Bideau & Learned-Miller, 2016). It includes 9 short video sequences in total that were extracted from YouTube videos and accompanying hand-labeled ground-truth masks on every 5^{th} frame. We also provide pseudo GT masks by bi-directional consistency check strategy (Teed & Deng, 2020) to enable further works to train on this small dataset.

COD10K. We train and evaluate all still image based methods on COD10K (Fan et al., 2021a). It is currently the largest COD dataset. It consists of 5,066 camouflaged images (3,040 for training, 2,026 for testing), which is divided into 5 super-classes and 69 sub-classes. This dataset also provides high-quality fine annotation, reaching the level of matting.

MoCA-Mask. The original Moving Camouflaged Animals (MoCA) Dataset (Lamdouar et al., 2020) includes 141 Video sequences. These sequences are collected from YouTube with mostly resolution 720×1280, and sampled at 24 fps, resulting in 37K frames. The dataset covers 67 kinds of animals moving in natural scenes. However, it contains a number of video sequence where animals are not well camouflaged and easily to be found. Also, the evaluation metrics only report animals located in a bounding box. Our goal is to build a comprehensive benchmark with more accurate evaluate criteria. To reach this, we reorganize the dataset as *MoCA-Mask*.

- **Remove Invalid Scenes.** We first select and exclude scenarios that animals are obvious and easy to identified from the background at our first glance. After cleaning the dataset, our new subset includes 87 video sequences, 22,939 frames in total.
- **Handy-craft Segmentation Masks.** For annotations, we further provide accurate handy-craft segmentation masks on every 5^{th} frame. Thus our handy-craft GT consists of two formats, that is 4,691 bounding box annotations as well as 4,691 pixel-level masks.
- **Generate Pseudo Masks.** We introduce a bidirectional optical flow-based strategy to generate our pseudo GT masks, refer to Section 3.4. Given a sequence of handy-craft GT with 5^{th} interval, we first estimate the forward and backward directions of optical flows and warp the GT with the corresponding optical flows. Then invalid pixels are eliminated by processing the bi-directional check.
- **Dataset Split.** The whole dataset is split into 71 sequences, 19,313 frames for training, and 16 sequences, 3,626 frames selected for testing purpose. The summary of each subsequence distribution could be found in the ***Appendix***.

4.2 BENCHMARKS

Settings. We compare with primarily top-performing approaches, both single image and video baselines. As the architectures, input resolution, modality, pre-processing and post-processing are all different, we try the best to conduct the comparison as fairly as possible. For single image baselines, we adopt the same data pre-processing with (Fan et al., 2020a; 2021a) for all the compared methods. Specifically, the input images are resized to 352×352, after random flip, random rotation and color enhance augmentation. In the training phrase, we apply random pepper noise on the GT images.

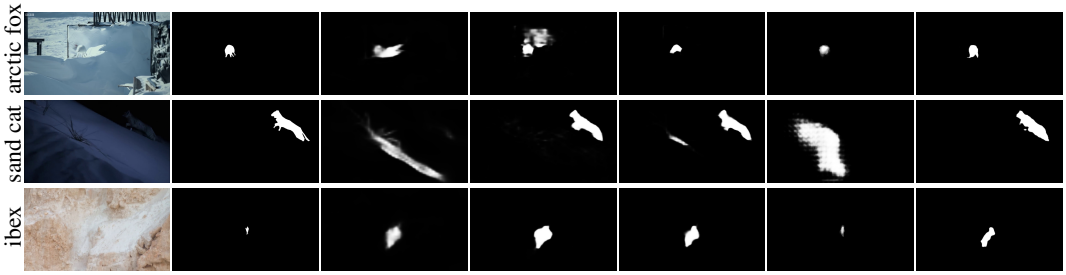


Figure 6: Qualitative results on our MoCA-Mask benchmark.

As EGNet (Zhao et al., 2019) requires extra edge / boundary information for training, we adopt the same pre-processing techniques in their paper to obtain the edge maps. This extra information could also be found in our reorganised version of MoCA-Mask dataset.

Most of video approaches, *i.e.,* PNS-Net (Ji et al., 2021a), RCRNet (Yan et al., 2019), apply the pipeline that train the static model first on still image datasets, and then optimize the enhanced architecture by interpolating temporal modules on video datasets. We follow this training setting and pre-train all methods on COD10K training set, except MotionGroup (Yang et al., 2021) which does not have a static model. Also, per our experimental experience, loading pre-trained weights on COD10K dataset could further improve the model performance on MoCA-Mask. Compared with COD10K image dataset, the video dataset MoCA-Mask is more challenging due to the camera motions, blurring images, small ratio of animals and their tiny body structures, such as slim torso / limbs. In some video sequences, the animals make up a very small proportion of the entire frame, which are extremely hardly can be identified, *i.e.,* ibex in Figure 6. Based upon considerations, we provide the results based on the following setting: (a) Training the models on COD10K; (b) Fine-tuning the models on MoCA-Mask, with pretrained weights on COD10K; (c) Evaluate the models on the whole CAD, the test set of COD10K, and MoCA-Mask dataset.

Performance on COD10K. We use the default training camouflaged images, and evaluate all the single image networks across all metrics on COD10k test set. Our still image network can be conducted easily by removing all the temporal correlation calculation, *i.e.,* short-term correlation pyramid and the spatial-temporal transformer module. Thus under the training setting on COD10k dataset, our network does not exploit any temporal information. As shown in Table 1, we observe that our network is better than other competitors.

Table 1: Quantitative results of single image baselines on COD10k dataset. Noting that all methods are trained using their original setting.

Models	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$E_\phi \uparrow$	$M \downarrow$
EGNet (Zhao et al., 2019)	0.749	0.557	0.780	0.048
BASNet (Qin et al., 2019)	0.788	0.646	0.857	0.044
CPD (Wu et al., 2019)	0.797	0.606	0.820	0.042
PraNet (Fan et al., 2020b)	0.812	0.681	0.884	0.036
SINet (Fan et al., 2020a)	0.796	0.660	0.876	0.039
SINet-v2 (Fan et al., 2021a)	0.816	0.685	0.890	0.035
PNS-Net (Ji et al., 2021a)	0.805	0.587	0.827	0.043
RCRNet (Yan et al., 2019)	0.795	0.614	0.829	0.043
SLT-Net - single (Ours)	**0.853**	**0.754**	**0.922**	**0.026**

Performance on MoCA-Mask. In Table 2, our short-term approach outperforms all the method significantly, notably by 10.88% on S_α over the best one in this evaluation, RCRNet (Yan et al.,

Table 2: Quantitative results on our MoCA-Mask with (w/) and without (w/o) our pseudo labels. Noting that MG (Yang et al., 2021) performs unsupervised learning that are trained without labels.

	MoCA-Mask w/o pseudo labels						w/ pseudo labels					
Models	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$E_\phi \uparrow$	$M \downarrow$	mDic	mIoU	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$E_\phi \uparrow$	$M \downarrow$	mDic	mIoU
EGNet (Zhao et al., 2019)	0.547	0.110	0.574	0.035	0.143	0.096	0.546	0.105	0.573	0.034	0.135	0.090
BASNet (Qin et al., 2019)	0.561	0.154	0.598	0.042	0.190	0.137	0.537	0.114	0.579	0.045	0.135	0.100
CPD (Wu et al., 2019)	0.561	0.121	0.613	0.041	0.162	0.113	0.550	0.117	0.613	0.038	0.147	0.104
PraNet (Fan et al., 2020b)	0.614	0.266	0.674	0.030	0.311	0.234	0.568	0.171	0.576	0.045	0.211	0.152
SINet (Fan et al., 2020a)	0.598	0.231	0.699	0.028	0.276	0.202	0.574	0.185	0.655	0.030	0.221	0.156
SINet-v2 (Fan et al., 2021a)	0.588	0.204	0.642	0.031	0.245	0.180	0.571	0.175	0.608	0.035	0.211	0.153
SLT-Net - single (Ours)	**0.631**	**0.291**	**0.700**	**0.030**	**0.349**	**0.264**	**0.648**	**0.330**	**0.748**	**0.025**	**0.375**	**0.289**
PNS-Net (Ji et al., 2021a)	0.544	0.097	0.510	0.033	0.121	0.101	0.576	0.134	0.562	0.038	0.189	0.133
RCRNet (Yan et al., 2019)	0.555	0.138	0.527	0.033	0.171	0.116	0.597	0.174	0.583	0.025	0.194	0.137
MG (Yang et al., 2021)	0.530	0.168	0.561	0.067	0.181	0.127	0.547	0.165	0.537	0.095	0.197	0.141
SLT-Net - short-term (Ours)	**0.628**	0.289	0.698	0.030	0.348	0.262	**0.662**	0.350	0.766	0.021	0.392	0.303
SLT-Net - long-term (Ours)	**0.628**	**0.292**	**0.704**	**0.028**	**0.351**	**0.264**	0.656	**0.357**	**0.785**	**0.021**	**0.397**	**0.310**

2019), and 89.19% on F_β^w metric over SINet (Fan et al., 2020a). We also provide the qualitative comparison of our method and other baselines in Figure 6. Our model can accurately locate and segment concealed objects in many challenging situations, such as objects with tinny torso or complex appearance textures, blur or abrupt motions.

Performance on CAD. In Table 3, we assess these different approaches by studying their cross-dataset generalization on CAD dataset. Again, the proposed network obtains the best performance in terms of six golden evaluation metrics, further demonstrating its robustness.

4.3 Ablation Studies

We perform ablation studies on the re-built MoCA-Mask dataset. In particular, we look into functionality analysis for our pseudo masks, short-term and long-term architectures.

Table 3: Quantitative results on CAD dataset.

Models	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$E_\phi \uparrow$	$M \downarrow$	mDic	mIoU
EGNet (Zhao et al., 2019)	0.619	0.298	0.666	0.044	0.324	0.243
BASNet (Qin et al., 2019)	0.639	0.349	0.773	0.054	0.393	0.293
CPD (Wu et al., 2019)	0.622	0.289	0.667	0.049	0.330	0.239
PraNet (Fan et al., 2020b)	0.629	0.352	0.763	0.042	0.378	0.290
SINet (Fan et al., 2020a)	0.636	0.346	0.775	0.041	0.381	0.283
SINet-v2 (Fan et al., 2021a)	0.653	0.382	0.762	0.039	0.413	0.318
PNS-Net (Ji et al., 2021a)	0.655	0.325	0.673	0.048	0.384	0.290
RCRNet (Yan et al., 2019)	0.627	0.287	0.666	0.048	0.309	0.229
MG (Yang et al., 2021)	0.594	0.336	0.691	0.059	0.368	0.268
SLT-Net - short-term (Ours)	0.696	0.471	0.827	0.031	0.484	0.392
SLT-Net - long-term (Ours)	**0.697**	**0.481**	**0.845**	**0.030**	**0.493**	**0.402**

Pseudo Masks. As shown in Table 2, the generated pseudo labels can largely improve the performance of all video approaches that require GT for supervision. For still image baselines, we interestingly found all the method with CNN encoders cannot exploit the pseudo masks well to further improve performance on MoCA-Mask test set. Although some approaches could reach a smaller MAE value, such as *i.e.,* CPD (Wu et al., 2019) and EGNet (Zhao et al., 2019), they failed in achieve higher results on other metrics. However, the single image version of our network could gain from the pseudo masks and achieve better performance.

Short-term *v.s.* Long-term Architecture. To analyze the effectiveness and efficiency of short-term and long-term architecture, we report their performances on MoCA-Mask dataset in Table 2 and CAD dataset in Table 3. On MoCA-Mask dataset, as shown in Table 2, the long-term architecture outperforms short-term architecture by an improved on F_β^w, E_ϕ, M, mDic, and mIoU, while only a slightly reduction on S_α. From the last two rows in Table 3, we observe that long-term architecture outperforms short-term architecture on all metrics.

Spatial-temporal Transformer *v.s.* ConvLSTM. We evaluate two different approaches for constructing long-term architecture, namely Spatial-temporal transformer, and ConvLSTM based refinement network. For the latter ConvLSTM network variant, we adopt a sequence model proposed by (Denton & Fergus, 2018) but modify the original VGG-style network for the CNN encoder and decoder with our transformer-style backbone network. From the Table 4, we could observe that the spatial-temporal transformer variant is more accurate than the ConvLSTM model but with a much lower number of parameters.

Table 4: Ablation studies of different long-term architectures. The input resolution is 256×448, and metrics are measured on MoCA-Mask test set.

Architecture Variant		**Achieved Network**				
Transformer	ConvLSTM	Params	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$E_\phi \uparrow$	$M \downarrow$
	✓	179.03 MB	0.651	0.348	0.767	**0.021**
✓		82.30 MB	**0.656**	**0.357**	**0.785**	**0.021**

5 Conclusion

We present a new SLT-Net framework for learning to segment concealed objects in video that includes i) a short-term module to implicitly capture motions between consecutive frames ii) a long-term module with a spatial-temporal transformer to enforce temporal consistency. To promote the development of this field, we rebuild a new dataset called MoCA-Mask with 87 high-quality video sequences including 22,939 frames in total. It is the largest-scale pixel-level annotated dataset which allows object-level benchmark in VCOD. Compared with existing cutting-edge baselines, our proposed network achieves fascinating results on three different camouflaged object benchmarks.

Acknowledgements. We thank the anonymous reviewers for the insightful comments on this paper.

REFERENCES

Sara Beery, Guanhang Wu, Vivek Rathod, Ronny Votel, and Jonathan Huang. Context r-cnn: Long term temporal context for per-camera object detection. In *CVPR*, 2020.

Pia Bideau and Erik Learned-Miller. It's moving! a probabilistic model for causal motion segmentation in moving camera videos. In *ECCV*, 2016.

Pia Bideau, Rakesh R Menon, and Erik Learned-Miller. Moa-net: self-supervised motion segmentation. In *ECCV Workshops*, 2018.

D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*, 2012.

Emily Denton and Rob Fergus. Stochastic video generation with a learned prior. In *ICML*, 2018.

Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A New Way to Evaluate Foreground Maps. In *ICCV*, 2017.

Deng-Ping Fan, Cheng Gong, Yang Cao, Bo Ren, Ming-Ming Cheng, and Ali Borji. Enhanced-alignment Measure for Binary Foreground Map Evaluation. In *IJCAI*, 2018.

Deng-Ping Fan, Wenguan Wang, Ming-Ming Cheng, and Jianbing Shen. Shifting more attention to video salient object detection. In *CVPR*, 2019.

Deng-Ping Fan, Ge-Peng Ji, Guolei Sun, Ming-Ming Cheng, Jianbing Shen, and Ling Shao. Camouflaged object detection. In *CVPR*, 2020a.

Deng-Ping Fan, Ge-Peng Ji, Tao Zhou, Geng Chen, Huazhu Fu, Jianbing Shen, and Ling Shao. Pranet: Parallel reverse attention network for polyp segmentation. In *MICCAI*, 2020b.

Deng-Ping Fan, Ge-Peng Ji, Ming-Ming Cheng, and Ling Shao. Concealed object detection. *IEEE TPAMI*, 2021a. doi: 10.1109/TPAMI.2021.3085766.

Deng-Ping Fan, Ge-Peng Ji, Xuebin Qin, and Ming-Ming Cheng. Cognitive vision inspired object segmentation metric and loss function. *SSI*, 2021b. doi: 10.1360/SSI-2020-0370.

Qibin Hou, Ming-Ming Cheng, Xiaowei Hu, Ali Borji, Zhuowen Tu, and Philip HS Torr. Deeply supervised salient object detection with short connections. In *CVPR*, 2017.

Suyog Dutt Jain, Bo Xiong, and Kristen Grauman. Fusionseg: Learning to combine motion and appearance for fully automatic segmentation of generic objects in videos. In *CVPR*, 2017.

Ge-Peng Ji, Yu-Cheng Chou, Deng-Ping Fan, Geng Chen, Huazhu Fu, Debesh Jha, and Ling Shao. Progressively normalized self-attention network for video polyp segmentation. In *MICCAI*, 2021a.

Ge-Peng Ji, Keren Fu, Zhe Wu, Deng-Ping Fan, Jianbing Shen, and Ling Shao. Full-duplex strategy for video object segmentation. In *ICCV*, 2021b.

Karthika Suresh Kumar and Aamer Abdul Rahman. Early detection of locust swarms using deep learning. In *MLCI*. 2021.

Hala Lamdouar, Charig Yang, Weidi Xie, and Andrew Zisserman. Betrayed by motion: Camouflaged object discovery via motion segmentation. In *ACCV*, 2020.

Trung-Nghia Le and Akihiro Sugimoto. Deeply supervised 3d recurrent fcn for salient object detection in videos. In *BMVC*, 2017.

Trung-Nghia Le and Akihiro Sugimoto. Video salient object detection using spatiotemporal deep features. *IEEE TIP*, 2018.

Trung-Nghia Le, Tam V. Nguyen, Zhongliang Nie, Minh-Triet Tran, and Akihiro Sugimoto. Anabranch network for camouflaged object segmentation. *CVIU*, 2019.

Guanbin Li and Yizhou Yu. Deep contrast learning for salient object detection. In *CVPR*, 2016.

Guanbin Li, Yuan Xie, Tianhao Wei, Keze Wang, and Liang Lin. Flow guided recurrent neural encoder for video salient object detection. In *CVPR*, 2018.

Ting Liu, Yao Zhao, Yunchao Wei, Yufeng Zhao, and Shikui Wei. Concealed object detection for activate millimeter wave image. *IEEE TIE*, 2019a.

Yi Liu, Qiang Zhang, Dingwen Zhang, and Jungong Han. Employing deep part-object relationships for salient object detection. In *ICCV*, 2019b.

Yunqiu Lv, Jing Zhang, Yuchao Dai, Aixuan Li, Bowen Liu, Nick Barnes, and Deng-Ping Fan. Simultaneously localize, segment and rank the camouflaged objects. In *CVPR*, 2021.

Ran Margolin, Lihi Zelnik-Manor, and Ayellet Tal. How to evaluate foreground maps? In *CVPR*, 2014.

Haiyang Mei, Ge-Peng Ji, Ziqi Wei, Xin Yang, Xiaopeng Wei, and Deng-Ping Fan. Camouflaged object segmentation with distraction mining. In *CVPR*, 2021.

Jeff Michels, Ashutosh Saxena, and Andrew Y Ng. High speed obstacle avoidance using monocular vision and reinforcement learning. In *ICML*, 2005.

Anestis Papazoglou and Vittorio Ferrari. Fast object segmentation in unconstrained video. In *ICCV*, 2013.

Xuebin Qin, Zichen Zhang, Chenyang Huang, Chao Gao, Masood Dehghan, and Martin Jagersand. Basnet: Boundary-aware salient object detection. In *CVPR*, 2019.

Anurag Ranjan, Varun Jampani, Lukas Balles, Kihwan Kim, Deqing Sun, Jonas Wulff, and Michael J Black. Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation. In *CVPR*, 2019.

Hongmei Song, Wenguan Wang, Sanyuan Zhao, Jianbing Shen, and Kin-Man Lam. Pyramid dilated deeper convlstm for video salient object detection. In *ECCV*, 2018.

Narayanan Sundaram, Thomas Brox, and Kurt Keutzer. Dense point trajectories by gpu-accelerated large displacement optical flow. In *ECCV*, 2010.

Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *ECCV*, 2020.

Pavel Tokmakov, Karteek Alahari, and Cordelia Schmid. Learning motion patterns in videos. In *CVPR*, 2017.

Linzhao Wang, Lijun Wang, Huchuan Lu, Pingping Zhang, and Xiang Ruan. Saliency detection with recurrent fully convolutional networks. In *ECCV*, 2016.

Wenguan Wang, Jianbing Shen, and Ling Shao. Video salient object detection via fully convolutional networks. *IEEE TIP*, 2017.

Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pvtv2: Improved baselines with pyramid vision transformer. *arXiv preprint arXiv:2106.13797*, 2021a.

Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *ICCV*, 2021b.

Jun Wei, Shuhui Wang, and Qingming Huang. F^3net: Fusion, feedback and focus for salient object detection. In *AAAI*, 2020.

Yu-Huan Wu, Shang-Hua Gao, Jie Mei, Jun Xu, Deng-Ping Fan, Rong-Guo Zhang, and Ming-Ming Cheng. Jcs: An explainable covid-19 diagnosis system by joint classification and segmentation. *IEEE TIP*, 2021.

Zhe Wu, Li Su, and Qingming Huang. Cascaded partial decoder for fast and accurate salient object detection. In *CVPR*, 2019.

Pengxiang Yan, Guanbin Li, Yuan Xie, Zhen Li, Chuan Wang, Tianshui Chen, and Liang Lin. Semi-supervised video salient object detection using pseudo-labels. In *ICCV*, 2019.

Charig Yang, Hala Lamdouar, Erika Lu, Andrew Zisserman, and Weidi Xie. Self-supervised video object segmentation by motion grouping. In *ICCV*, 2021.

Zhao Yang, Qiang Wang, Luca Bertinetto, Weiming Hu, Song Bai, and Philip HS Torr. Anchor diffusion for unsupervised video object segmentation. In *CVPR*, 2019.

Qiang Zhai, Xin Li, Fan Yang, Chenglizhao Chen, Hong Cheng, and Deng-Ping Fan. Mutual graph learning for camouflaged object detection. In *CVPR*, 2021.

Jia-Xing Zhao, Jiang-Jiang Liu, Deng-Ping Fan, Yang Cao, Jufeng Yang, and Ming-Ming Cheng. Egnet:edge guidance network for salient object detection. In *ICCV*, 2019.

Xiaoqi Zhao, Youwei Pang, Lihe Zhang, Huchuan Lu, and Lei Zhang. Suppress and balance: A simple gated network for salient object detection. In *ECCV*, 2020.

Mingchen Zhuge, Deng-Ping Fan, Nian Liu, Dingwen Zhang, Dong Xu, and Ling Shao. Salient object detection via integrity learning. *arXiv preprint arXiv:2101.07663*, 2021.

A APPENDIX

Short-term Correlation Pyramid Details We aggregate the channel information by a convolution operation $\phi(\cdot)$ and obtain a refined feature map $\phi(\mathbf{I}_{t+1}) \in \mathbb{R}^{C\times H'\times W'}$. Specifically, the aggregated features $f'_{t\leftarrow t+1} = \rho(\mathbf{I}_{t\leftarrow t+1}) \in \mathbb{R}^{C\times H'\times W'}$ was computed as follows:

$$\rho(\mathbf{I}_{t\leftarrow t+1}) = \tilde{\mathbf{C}}(\mathbf{I}_t, \mathbf{I}_{t+1})\phi(\mathbf{I}_{t+1}). \tag{6}$$

To enable the network to learn detailed information, a correlation pyramid $\mathbf{C}^i, i \in \{2,3,4\}$ is construct by incorporating multi-scale features. Thus for a sequence of frame features $\{\mathcal{F}_\theta(\mathbf{I}_t), \mathcal{F}_\theta(\mathbf{I}_{t+1})\} \in \mathbb{R}^{C\times H/2^{i+1}\times W/2^{i+1}}$, our short-term correlation pyramid can be denoted as $\mathbf{C}^i(\mathbf{I}_t, \mathbf{I}_{t+1}) \in \mathbb{R}^{H/2^{i+1}\times W/2^{i+1}\times H/2^{i+1}\times W/2^{i+1}}$. It outputs an aggregated feature map $f'^{(i)}_{t\leftarrow t+1}(\mathbf{I}_{t\leftarrow t+1})$ at the pyramid scale $i, i \in \{2,3,4\}$, which has the same dimension as the reference frame feature $\mathcal{F}_\theta(\mathbf{I}_t)$. We also repeat the correlative aggregation once on every other neighboring frame. In this way, we obtain aggregated feature maps $f'^{(i)}_{t\leftarrow t+1}(\mathbf{I}_{t\leftarrow t+2})$.

Training Details We implement both long-term and short-term architecture in PyTorch. The input images are resized to 352×352. We train the short-term architecture with a batch size of 8 on an NVIDIA V100 GPU and use Adam optimizer with initial learning rate of 1e-4, decreasing every 50k iterations. For the long-term optimization, our model takes 10 frames as the input at one time with the frame sampling rate 1. For our pseudo ground-truth generation, we exploit RAFT (Teed & Deng, 2020) as the optical flow estimation module and pre-trained weights on Sintel dataset (Butler et al., 2012).

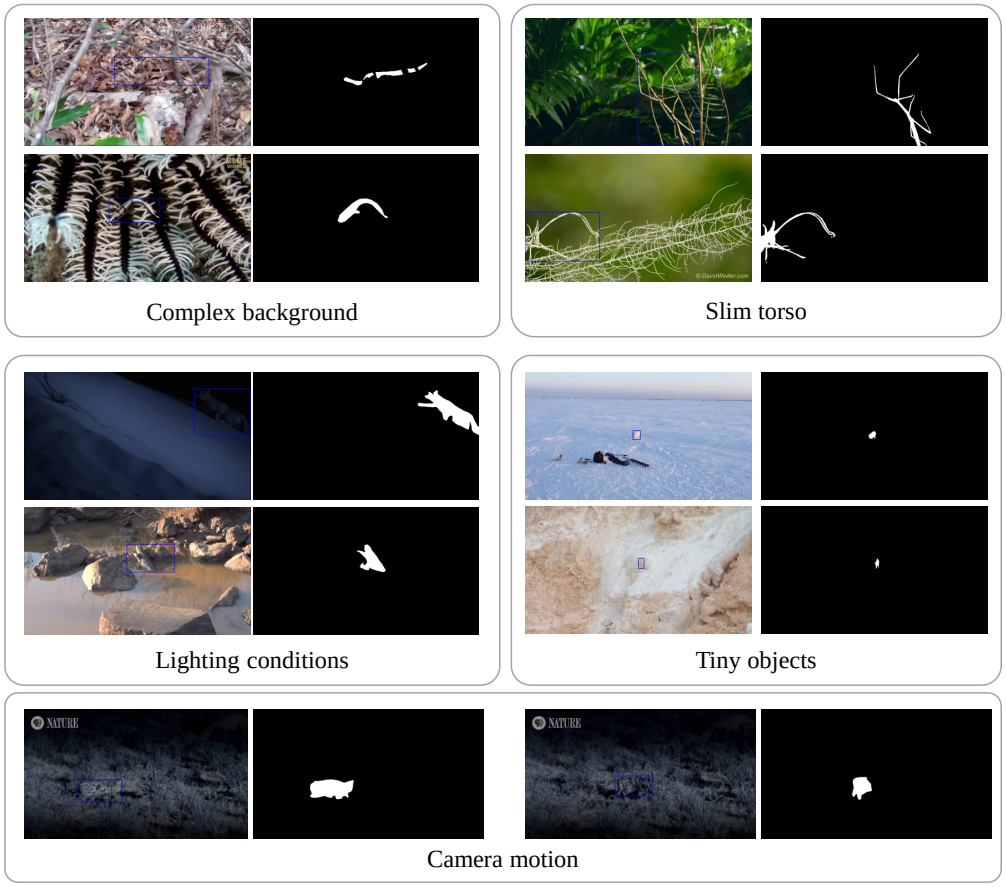


Figure 7: Representative samples from MoCA-Mask. The dataset is quite challenging including diverse scenes, suash as various lighting conditions, *i.e.,* dark and sunny, complex background, camera motions, small ratio of animals and tiny body structures, such as slim torso /limbs.

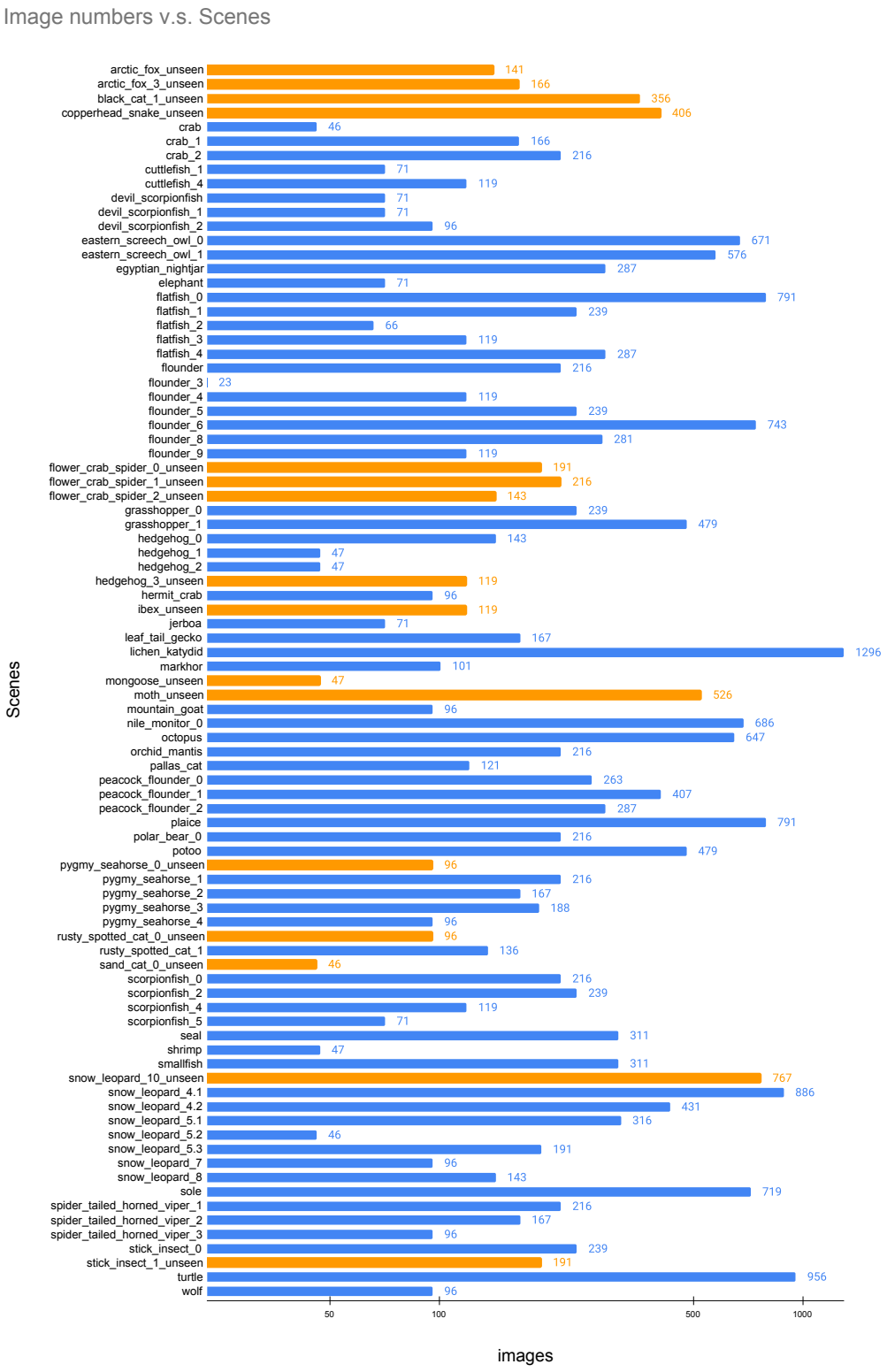


Figure 8: Summary for training and test set distribution. Our MoCA-Mask dataset includes 87 video sequences in total, in which 16 sequences were tagged as "unknow" (colored in orange). This split is used to validate the sensitivity of different models on novel samples. Zoom-in for details.

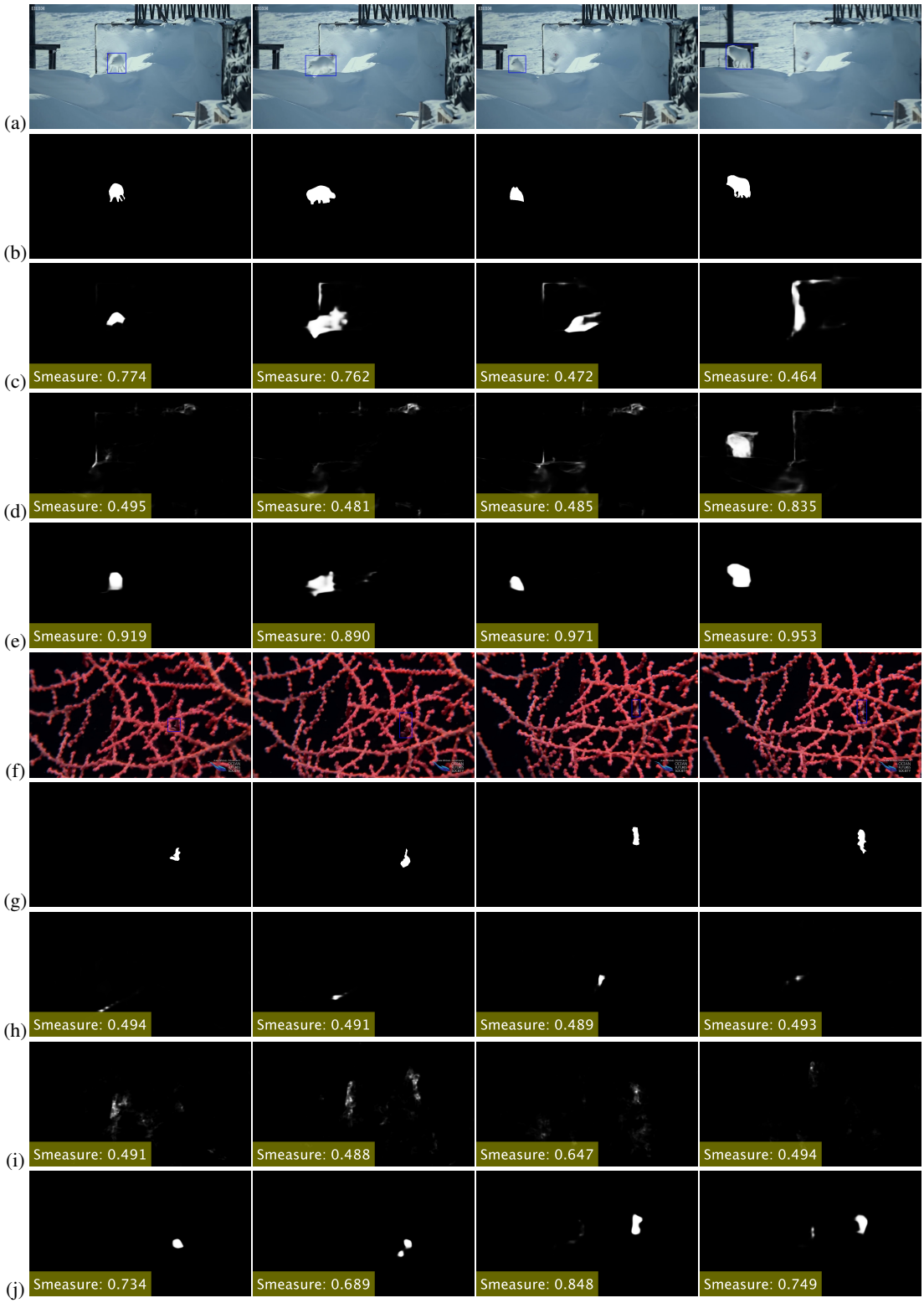


Figure 9: Comparison of our proposed network with two top-performing baselines on MoCA-Mask test dataset. Example squences of each row means: (a) (f) Frames, (b) (g) GT, (c) (h) SINet (Fan et al., 2020a), (d) (i) RCRNet (Yan et al., 2019), (e) (j) SLT-Net (Ours).