Thanks to the reviewers for their thoughtful feedback.

R1: "no explanation on technicalities". We agree wholeheartedly with R1 that it is important to explain the technicalities. We explained them in Introduction and Sec. 3. The merging in Fig. 4 is explained in Lines (**L**) **303-312**. It is essential for integrating the features from foreground (FG) and background (BG). We will explain more in the revision.

R1: "motivation of using correlation". The idea of correlation is widely used in object tracking. In SiamFC, correlation is computed between features of current frame and template frame to locate the position of the whole object using the produced similarity map. In RANet, to predict each pixel of object(s) for segmentation, we need pixel-level similarity maps by calculating the correlation between features of each pixel in template frame and that in current frame.

R1: "motivation of attention module". We provided the motivation of our RAM module in **L112-120** and **L263-269**. In short, RAM is proposed mainly to "tackle the dynamic channel size of FG/BG similarity features" (**L266**). It has little relationship with manifold, feature diversity, *etc*. We explain "The discrepancy ..." in **L372-376** of Sec. 3.2. It alleviates matching since it provides abundant and ordered features for segmentation. In Tab. 4 of the main paper, $w/o\ Ranking$ provides dis-ordered features, while $Maximun$ losses most information (**L743-747**). Both variants perform worse than RAM for our RANet on VOS task.

R1: "temporal propagation". We firstly described it in **L82-84**, as a trivial technique used in propagation based VOS methods. For our method, we first mention it in **L104-107**, and then take it into account in **L300-308** of Sec. 3.1.

R2+R3: "performance on multiple objects". The discrepancy on the results of our RANet for multi-object and single-object VOS are attributed to the limitation of the employed pixel-level similarity on distinguishing similar but different object instances, especially when huge appearance change and occlusion occurs. However, as shown in Tab. 1, the performance of RANet can be improved by fine-tuning on YouTube-VOS dataset. It obtains the same $\mathcal{J}$ mean (55.2) as that of FEELVOS on DAVIS$_{17}$-$testdev$, at a much faster speed. RANet also performs better than RGMP (51.3), which heavily rely on synthetic training data.

R3: "saliency". Since the proposed RANet relies on matching results instead of deep features for object locating, it does not necessarily rely on saliency. Besides, the off-line version of OSVOS and OnAVOS rely solely on saliency, but achieved bad performance on separating different FG objects (pleaser refer to the Tab. 3 in the main paper).

R2: "comparison with PReMVOS and FEELVOS". We totally agree that comparing with FEELVOS is very interesting. The ideas presented in FEELVOS are very insightful, and can potentially boost our RANet. For instance, the local matching will be helpful in solving the miss-matching problem between similar instances, and pre-training on semantic segmentation will also boost the accuracy. PReMVOS is powerful on VOS by using online learning (OL), mask proposals and ReID techniques, but very time-consuming (about 1000 times of ours). We will add the results and discussion of PReMVOS and FEELVOS in the revision.

Method	OL	DAVIS$_{17}$-val		DAVIS$_{17}$-testdev	
		$\mathcal{J}\&\mathcal{F}\uparrow$	$\mathcal{J}$ Mean$\uparrow$	$\mathcal{J}\&\mathcal{F}\uparrow$	$\mathcal{J}$ Mean$\uparrow$
PReMVOS (Y)	✓	**77.8**	**73.9**	**71.6**	**67.5**
FEELVOS	✗	69.1	65.9	54.4	51.2
RANet	✗	65.4	63.0	51.5	50.0
FEELVOS (Y)	✗	71.5	69.1	57.8	55.2
RANet (Y)	✗	68.4	66.0	57.2	55.2

Table 1. Comparison of different methods on multiple object VOS. The methods trained on YouTube-VOS are appended by "(Y)".

R2: "experiments on YouTube-VOS". We are struggling to fine-tune our RANet on YouTube-VOS, and will release the results later. The difficulties are: it is unclear in different methods on how to deal with the multiple reference frames in many videos of YouTube-VOS. The temporal stability of the results are studied in Tabs. 1 and 2 of the *Supp. File*.

R2: "Writing and others". Thank you for your enthusiastic suggestions. We will definitely improve the writing and other aspects (e.g., references and OL section) in revision.

R3: "evaluation is weak". We agree that diverse datasets are needed. Therefore, we evaluate our RANet not only on DAVIS 16-val (20 videos) and 17-val (30 videos), but also on 16-$trainval$ (50 videos) and 17-$testdev$ (30 videos). RANet can perform well without being trained on the DAVIS dataset: it still achieves state-of-the-art performance even being trained only with static images (please see Tab. 2 of the main paper). Hence, it can generalize well. We will also provide more results on other benchmarks.

R3: "network design is complicated". We integrate the max-pooling and two-layer network to provide global and local attention capacity for the proposed RAM module. Deleting either path will result accuracy drops on VOS.

R3: "technical details of ranking". As described in **L334, L422**, we do ranking by permuting channels. It costs little time by using PyTorch's *sort* and *index_select* functions (this operation is also used in the efficient ShuffleNet).

R3:"extra pretraining datasets". Many methods (CNIM, Masktrack, *etc*.) utilized Deeplab as backbone which is pretrained on larger MS-COCO and Pascal datasets. Compared with these methods, our RANet adopted smaller saliency datasets (including about 15K images) for pretraining, and used a standard ResNet as backbone. Moreover, MaskTrack also used these saliency datasets for pretraining, and RGMP used synthetic training data from the Pascal dataset.

R3: "Others". **1)** The last three columns of Tab 1 is boundary accuracy, which can be referred in [37]; **2)** "fine-grained" means extra accurate; **3)** The structure of the decoder is shown in Sec. 2 of the *Supp. File*; **4)** We use ℓ_2 normalization before computing similarity; **5)** As mentioned in Sec. 2 of the *Supp. File*, we use W=54, H=30, W_0=27 and H_0=15 in our RANet for DAVIS datasets.