We thank all reviewers for their efforts and positive comments.

To Review #1: Thank you for the comments and we will follow the suggestions to do more experiments and analyses.

To Review #2: Thank you very much for confirming our contributions.

Q1) [Strong priors]: “geometry-consistency" is not a manually designed prior for specific scenes. It is indeed an underlying constraint of Structure-from-Motion or Multi-View Geometry [3]. A well-known method that enforces geometry-consistency is Bundle Adjustment. BA is the core method of Visual SLAM and SfM, and it is widely used in general scenes. Here, unsupervised depth learning relies on the theory of Structure-from-Motion, as discussed in [21], so the baseline research on this problem is called SfMLearner [6]. CC [10] and all other related works share the same theory for unsupervised depth learning. Therefore, compared with them, our method does not require additional strong priors. Our contribution is more deeply leveraging SfM theory in unsupervised depth learning.

Q2) [Improvement over SOTA]: Yes, although the performance is just slightly better than CC [10], our contribution is solving scale-inconsistency issue and a more efficient framework. More advantages like efficiency would be added.

To Review #3: Thank you for your comments.

Q3) [How much cost is saving]: We measure the time taken for each iteration consisting of forward and backward pass using a batch size of 4. Below shows the results on 832×256 images. CC [10] needs train 3 parts iteratively, including (DepthNet, PoseNet), FlowNet, and MaskNet. Our method only trains (DepthNet, PoseNet) for 200K iterations. In total, CC takes about **7 days** for training, while our method takes **32h27m10s**. The results of CC [10] are reported by authors. Both methods are tested on a single 16GB Tesla V100 GPU.

Table 1: Training time and the number of model parameters for each network

	CC [10]			Ours
Network	(DepthNet, PoseNet)	FlowNet	MaskNet	(DepthNet, PoseNet)
Time	0.96s	1.32s	0.48s	0.55s
Parameter Numbers	(80.88M, 2.18M)	39.28M	5.22M	(80.88M, 1.59M)

Moreover, we show the time consumption for obtaining mask. CC involves FlowNet and MaskNet, and we use GC+M. Time for one iteration is shown below. FlowNet is slow due to correlation calculation, which is time-consuming. Ours (GC+M) is fast because only several basic operations such as subtraction and division are involved for mask computation, as indicated in Eqn 5-8. Note that the most time-consuming part (reprojection) in our method has been done when computing photometric loss, which can be reused in computing mask.

Table 2: Time to obtaining mask on a single 8GB RTX 2080.

	CC [10]		Ours
Method	FlowNet (Forward)	MaskNet (Forward)	GC+M
Time	42.73ms	16.94ms	0.0002ms

Q4) [Moving objects, Static scene assumption]: Static assumption (or called rigid assumption) is the prerequisite of perspective projection [3], i.e., depth and pose based projection. The perspective projection allows for photometric loss in unsupervised depth learning. Therefore, photometric loss on dynamic scenes (moving objects) are meaningless and cause false supervisions. The rigid assumption and perspective projection were discussed in Mult-View Geometry [3]. Besides, GeoNet[8], DF-Net[9], and CC[10] introduced the adverse effect of moving objects for unsupervised depth learning and used optical flow networks to localize dynamic regions.

Q5) [Scale ambiguity, Inconsistency issue]: Scale ambiguity is a well-recognized issue in pure monocular systems. It refers to the issue that the scale that aligns the estimated scene to the real-world scene is unknown. It can be resolved by introducing another devices (stereo camera or range sensor) or relying on specific strong priors. Indeed, all related works (from SfMLearner to CC, and ORB-SLAM) estimate relative depths instead of real depths due to scale ambiguity. Inconsistency issue arises due to the fact that SfMLearner and related methods take one snippet (3 or 5 frames) as input during tracking and do not consider the consistency between different snippets. As a result, the scaling factors that align the estimated camera poses (translation) to real-world varies for each snippet, i.e. inconsistent scales. Therefore, related works evaluate pose estimation results using 5-frame pose metrics [6] instead of using more general visual odometry evaluation metrics. In our method, we enforce consistent scales by enforcing depth consistency across snippets, which further allows pose consistency across snippets.

Q6) [Mask mitigates the issue of moving objects]: Fig 2 demonstrates that our mask can detect moving objects and occlusions. More visual results are in supplementary. Tab 1 demonstrates that using mask can boost performance.

Q7) [Compare with [7]]: Have compared in Tab 3. Regarding AbsRel, it is 0.137 vs 0.163 in (K) and 0.128 vs 0.159 in (CS+K).