Thanks for the valuable comments. We are happy that most reviewers recognized the importance and novelty of our work. Below we clarify each question and we hope reviewers can raise their scores based on the responses.

[R1 & R3: Reproducibility] In Section 4 (L194 $\sim$ L205), we have provided the detailed experiment settings. We do not use any tricks but follow the classic settings for image classification and 3D recognition. Our idea is very easy to implement, we will release our code immediately upon acceptance.

[R1 & R3: Training cost and training strategy.] Though the passport-aware and passport-free branch are trained alternatively by default, experimental results show that they ***can also be trained simultaneously*** with similar performance. For the training cost, it indeed depends on the ratio of the passport-aware branch activated in every training epoch. The default ratio value is 50%, i.e., train 1 iteration passport-aware branch after training every 1 iteration passport-free branch. In this case, the theoretical computation cost will be 2x. However, ***we find it feasible to use a lower ratio for the passport-aware branch with very comparable performance***. For example, when the ratio is 10%, i.e.,train 1-iteration passport-aware branch after training every 9 iterations passport-free branch, its extra computation cost is only about 10%. Under this setting, take PointNet on ShapeNet dataset for example, the verification accuracy is almost unchanged (from 99.47% to 99.31%) while the model deployment performance is even slightly better (from 99.31% to 99.36%). ***More importantly, this will not introduce any extra cost for deployment***.

[R2:Motivation of the transformation in eq3?] Though our transformation formulation looks similar to that in SE, ***the underlying consideration is totally different***. In SE, the transformation is to learn a channel-wise attention to enhance the feature representation. But in our method, our motivations are from another two aspects: 1) The input of passport-aware transformation wp_γ, wp_β ($wp_k = W_c \otimes p_k$) share the same convolution kernel W_c with the convolutional feature x, considering the physical meaning and value difference between p_γ, p_β and x, the non-linear transformation is used to ***remap the*** wp_γ, wp_β ***to meaningful transformation parameters*** γ_1, β_1, we find it can improve the training stability and performance (Table 6 and Figure 3). 2) Using this non-linear transformation ***can help increase the difficulty of passport reverse-engineering***, thus enhancing the robustness (Table 4,5).

[R2: Difference between (eq2,eq3) *vs* eq4, extra complexity?] Compared to the baseline eq4, our formulation (eq2, eq3) innovatively decouples the passport-free and passport-aware branch and uses an extra non-linear transformation, which are both the keys of our stronger generalization ability and performance. Compared to baseline [17], we find the empirical training time is almost the same. More importantly, it does not introduce any extra parameter or computation into the deployed model but only during forensics and training.

[R2: Comparison with transformation based normalization methods.] Thanks for your suggestion. We tried the famous conditional normalization layer SPADE in “Semantic Image Synthesis with Spatially-Adaptive Normalization". Considering the conditional input in our task is the one-dimension transformed passport wp_γ, wp_β, we replace the convolution layer used in SPADE with fc layer. Then it can be viewed as a special case (i.e., only the nonlinear transform in eq3) of our method without the decoupled design in eq2. Like [17], it needs to replace BN with GN, otherwise it will produce very bad results (e.g., 21.99/1.90 for ResNet-18 on CIFAR10/CIFAR 100 (ours 94.25/74.40)) .

[R3: Is two-step verification necessary?] Yes. As described in **L36-42, L85-88 and [17]**, though trigger-set-based methods support black-box verification, they are fragile to ambiguity attacks. In details, the attacker can forge another set of trigger images to make the ownership claim ambiguous. In contrast, our passport based method can resist ambiguity attack but cannot support black-box (remote) verification (**L178-182**). ***Therefore, using the two-step verification can combine the advantages of these two methods, i.e., not only support remote verification but resist ambiguity attack***.

R3: Trigger-set-based details. We adopt a similar setting as the trigger-set based method [10]. We use about 100 images/points not belonging to the target dataset as the trigger set for all tasks. Empirically, we find adding such trigger sets into training only slightly affects the target model performance. For instance, with ResNet-18 on the CIFAR10, the model performance only drops slightly to 94.25% from 94.36%. We will add such analysis to the final version.

[R3:Broader impact.] As R5 and L293-297 described, IP protection for deep models is an important but currently under-researched area. Our method can help protect deep business models from illegal distribution or usage.

[R5: Key differences with the baseline [17]]. As described in section “relationship to [17]" (L151-162), the formulation (Eq4) of [17] is a special case of our formulation (Eq2). Compared to [17], we have two key differences: 1) decoupling passport-aware branch from passport-free branch; 2) learnable non-linear affine transformation. We emphasize these two points are both important and non-trivial. The former point can significantly boost the generalization ability for passport-based IP protection methods without any architecture change (e.g., BN to GN). The latter point not only improves training stability and performance (shown in Figure 3 and Table 6), but also significantly enhances the robustness against ambiguity attack (shown in Table 4, 5) compared with the non-learnable parameters used in [17].

[R5: Typos.] Thanks, we will improve it in the final version.