We want to thank all reviewers for their constructive feedback. Here, we reply to all questions and present more results (on **COCO** in Tab. 1 and **after 800 epochs** in Tab. 2) that further support the efficacy of the proposed approach.

Summary of contributions. Despite the multitude of SSL papers during the months before *and after* the NeurIPS submission deadline, we find that our contributions still stand, and we are glad that reviewers also generally agree that: a) **hard negatives in contrastive SSL** are under-explored beyond MoCHi and our **empirical analysis** and oracle experiments are interesting (**R1**, **R2**, **R3**) and may inspire future research (**R2**); b) **mixing for hard negatives** is novel (**R1**, **R2**, **R3**) and produces **consistent gains** (**R1**, **R3**) over the state-of-the-art (SoTA) MoCO-v2 on 3 datasets (COCO, VOC, Im-100) and 3 tasks; c) We report results that **set a new SoTA** also for shorter (100 epoch) pre-training. We are also glad that reviewers **R1**, **R3** and **R4** found our paper clearly written and easy to read.

Table 1: Object detection and instance segmentation results **on COCO with the** $\times 1$ **training schedule** (C4 backbone).

Pre-train	Epochs	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
Supervised [13]		38.2	58.2	41.6	33.3	54.7	35.2
MoCo [13]	200	38.5	58.3	41.6	33.6	54.8	35.6
MoCo (1B image train) [13]	200	39.1	58.7	42.2	34.1	55.4	36.4
InfoMin Aug. [28]	200	39.0	58.5	42.0	34.1	55.2	36.3
MoCo-v2 [6]	200	39.0 (±0.1)	58.6 (±0.1)	41.9(±0.3)	34.2 (±0.1)	55.4 (±0.1)	36.2 (±0.2)
+ MoCHi (256, 512, 0)	200	39.2 (±0.1) (↑0.2)	58.8 (±0.1) (↑0.2)	42.4 (±0.2) (↑0.5)	34.4 (±0.1) (↑0.2)	55.6 (±0.1) (↑0.2)	36.7 (±0.1) (↑0.5)
+ MoCHi (128, 1024, 512)	200	39.2 (±0.1) (↑0.2)	58.9 (±0.2) (↑0.3)	42.4 (±0.3) (↑0.5)	34.3 (±0.1) (↑0.2)	55.5 (±0.1) (↑0.1)	36.6 (±0.1) (↑0.4)
+ MoCHi (512, 1024, 512)	200	**39.4** (±0.1) (↑**0.4**)	**59.0** (±0.1) (↑**0.4**)	**42.7** (±0.1) (↑**0.8**)	**34.5** (±0.0) (↑**0.3**)	**55.7** (±0.2) (↑**0.3**)	**36.7** (±0.1) (↑**0.5**)
MoCo-v2 [6]	100	37.0 (±0.1)	56.5 (±0.3)	39.8 (±0.1)	32.7 (±0.1)	53.3 (±0.2)	34.3 (±0.1)
+ MoCHi (256, 512, 0)	100	37.5 (±0.1) (↑0.5)	57.0 (±0.1) (↑0.5)	40.5 (±0.2) (↑0.7)	33.0 (±0.1) (↑0.3)	53.9 (±0.2) (↑0.6)	34.9 (±0.1) (↑0.6)
+ MoCHi (128, 1024, 512)	100	**37.8** (±0.1) (↑**0.8**)	**57.2** (±0.0) (↑**0.7**)	**40.8** (±0.2) (↑**1.0**)	**33.2** (±0.0) (↑**0.5**)	54.0 (±0.2) (↑**0.7**)	35.4 (±0.1) (↑**1.1**)

Does MoCHi learn faster? (R1, R2) Yes! To further justify the learning speed of MoCHi beyond the training curves of Fig.2, we report results on VOC (Fig. 4a) and also COCO (Tab. 1) after 100 epoch pre-training, *i.e.* half of the standard 200 epochs. MoCHi reaches *performance similar to supervised pre-training (33.2) in 100 epochs* for instance segmentation on COCO. We further measured our computational overhead in terms of wall-clock time (**R2**) for runs with/out MoCHi and found that MoCHi training was approx. 5-25% slower (different params/machines/loads).

Are the gains significant? (R2, R3, R4) To further support MoCHi's efficacy, in Tab. 1 we report results on COCO (requested by **R2**), that further show MoCHi achieving *consistent gains* over SoTA. We see that a) MoCHi outperforms the SoTA method of [28] by ~0.5%, supervised pre-training by over 1%, and has higher gains over MoCo-v2 when training for only 100 epochs. On VOC we see from Fig.4a that a) gains are robust to different configurations b) gains are larger (~1+%) for transferring after only 100 epoch training and c) from Tab. 2 we see that *gains persist after longer training*; we report a new SoTA performance for transfer learning on VOC.

Table 2: Results after training for 800 epochs.

Method	IN-1k		VOC 2007	
	Top1	AP_{50}	AP	AP_{75}
Supervised [13]	76.1	81.3	53.5	58.8
MoCo-v2 [6]*	69.0	82.7 (±0.1)	56.8 (±0.2)	63.9 (±0.7)
+ MoCHi (128, 1024, 512)	68.7	**83.3** (±0.1)	**57.3** (±0.2)	**64.2** (±0.4)

Is MoCHi complementary to iMix [25] and [28]? (R2, R3) We first want to respectfully remind the reviewers that both are *unpublished*, concurrent works. [25] mixes *in image space* and does not deal with neither hardness nor the negatives. Focusing on hard negatives allows MoCHi to a) show significant gains on transfer learning *after only 100 epochs*); b) achieve significantly higher performance than [25] for longer training (Tab. 1 in the supplementary and Tab. 2). Our method performs on par with (and better than) [28] on VOC (and on COCO). We believe (but didn't have time to verify) that better positive pairs (*e.g.* from [28]) are indeed complementary to MoCHi.

Are the two mixing methods complementary? (R2, R3). We agree that Fig. 3b gives mixed messages; this is because it shows the *first* run per combination, not the mean (we didn't have multiple runs for all combinations, but only the top ones). We will replace all numbers with means and variance (numbers we now have) and it will then be more clear, together with the COCO and VOC results, that both are needed to get top performance.

On missing works from metric learning (R4). We thank the reviewer, these are indeed related works that we will cite and discuss. The two paper mentioned by **R4**, similar to the even newer [15] from CVPR 2020, operate on metric learning losses and in a supervised setting (see l97-101 for differences to [15]). We instead focus on the loss of Eq.(1), have no class labels, and exploit the *memory*, something not present in metric learning works. What is more, both works mentioned by **R4** require a generator, *i.e.* extra parameters and loss terms that need to be optimized; MoCHi has no added parameters and further goes beyond triplets, by mixing negatives from the memory.

On MoCHi being "simple and trivial" (R3). We agree that MoCHi is a **simple** method; yet we argue that this is not a weakness in itself, but a desirable characteristic given that it gives consistent gains. We also argue that it is **novel** in the realm of SSL, as **R1**, **R2** and also **R3** themselves do. However, we respectfully disagree that it is **trivial**. We hope we clarified the other two points listed as weaknesses by **R3** (novelty over the concurrent [25] and complementarity of the mixing methods), but we cannot really rebut "trivial", a claim that can be neither properly justified, nor rebutted.

Mixing the query as rescaling of the loss (R1). Thanks, this is correct; we will add this intuition in the text.

Other points by R2. a) *Probability in Fig. 2(a) goes over 1*: There is a scaling factor of 1e-2 (top left corner); b) *Paper organization/writing*: We agree that parts of the text can be more clearly presented and structured; we will reorganize and revise all Figs; c) *L141-150*: a great point; we will run an experiment to directly evaluate and rephrase.