Supplementary Material of Towards Backward-Compatible Representation Learning

1. Implementation Details

In Section 2.5, we describe how to use the influence loss on newly added training data for BCT to 1) compute synthesized classifier weights with the old model, and 2) use knowledge distillation.

When we have a newly added training example whose class is not in the old embedding training set, we feed the new image into the old model ϕ_{old} and old classifier $w_{\text{c old}}$ to obtain the classifier responses. Then, we can provide supervision signal to the new model ϕ_{new} for this image by feeding the new model's embedding to the old classifier $w_{\text{c old}}$ and compute the knowledge distillation loss (cross-entropy with temperature-modulated SoftMax) between the two response vectors as the influence loss. Because we are using the cosine margin loss [4] in the experiments which also has a temprature parameter, we set the temperature parameter in knowledge distillation to the same as the one in the cosine margin loss, which is 32.

2. Partial Backfilling

Partial backfilling happens when only a part of the gallery classes have been processed by the new model ϕ_{new}. We test whether queries from the backward compatible new model $\phi_{new-\beta}$ can work with partially backfilled gallery sets. In Fig. 1, we illustrate the search accuracy on gallery sets of different backfill ratios. As higher percentages of the gallery set are backfilled, search accuracy grow proportionally towards the accuracy of the fully backfilled case. This suggests that one can upgrade to a new model, immediately benefiting from the improved accuracy, and optionally backfill the old gallery gradually in the background until paragon performance is achieved.

3. Detailed Benchmark Results

Due to space limitations, in the main paper we only report one specific operating point for each metric in the evaluation, *e.g.*, $\text{TAR@FAR} = 10^{-4}$ for face verification and $\text{TPIR@FPIR} = 10^{-2}$ for face identification. Here we report results at additional operating points on the IJB-C [2] benchmark. In Table 3, we show the performance of different compared baselines and our proposed method. In Table 2 and Table 4, we show the performance of extensions of our proposed backward-compatible training process. In Table 2, we illustrate extensions of BCT to different model depths, feature dimensions and supervision losses. In Table 4, we show extensions to multi-model compatibility towards sequential updating.

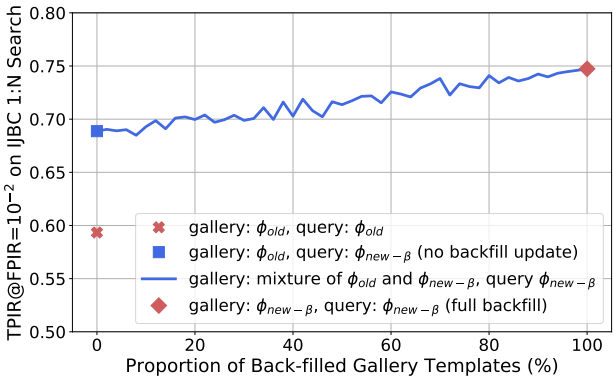

Figure 1: The curve for face search accuracy vs. backfill proportions. We gradually backfill the gallery set from 0% classes (same as the backward compatibility test or no-backfill update) to 100% (fully backfill or the paragon setting). Face search accuracy is measured between every 2% of partial backfill. The red cross shows the face search accuracy of the old model. The red diamonds marks the face search accuracy of the new embedding model.

References

[1] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017. 3

[2] Brianna Maze, Jocelyn Adams, James A Duncan, Nathan Kalka, Tim Miller, Charles Otto, Anil K Jain, W Tyler Niggel, Janet Anderson, Jordan Cheney, et al. Iarpa janus benchmark-c: Face dataset and protocol. In *2018 International Conference on Biometrics (ICB)*, pages 158–165. IEEE, 2018. 1, 3

[3] Feng Wang, Xiang Xiang, Jian Cheng, and Alan Loddon Yuille. Normface: l 2 hypersphere embedding for face ver-

	Continual Learning	Domain Adaptation	BCRL
Access to all old model parameters	Yes	Yes	Not required
Access to old training data	Not required	Yes	Yes
Re-processing of test data?	Yes	Yes	Not required
Consistent output	Yes	Not required	Yes
Compatible representation	Not required	Not required	Yes

Table 1: The differences between backward compatible representation learning (BCRL) , continual Learning, and domain adaptation. In this table, we list several features of the task each problem deals with to illustrate the difference.

New Model	Old Model	Training Data Usage	Feat. Dim.	Model Arc.	Classifier	Additional Loss
ϕ_{old}	-	50%	128	ResNet-101	Cosine Margin	-
$\phi_{\text{new}-\beta}^{\text{R152}}$	ϕ_{old}	100%	128	ResNet-152	Cosine Margin	Influence loss on $\mathcal{T}_{\text{old}}$
$\phi_{\text{new}-\beta}^{\text{R152+256D}}$	ϕ_{old}	100%	256	ResNet-152	Cosine Margin	Influence loss on $\mathcal{T}_{\text{old}}$
$\phi_{\text{new}-\beta}^{\text{ReLU}}$	ϕ_{old}	100%	128	ResNet-152	Cosine Margin	Influence loss on $\mathcal{T}_{\text{old}}$
$\phi_{\text{old}}^{\text{NS}}$	-	100%	128	ResNet-101	Norm-Softmax	-
$\phi_{\text{new}-\beta}^{\text{Cos}-\text{NS}}$	$\phi_{\text{old}}^{\text{NS}}$	100%	128	ResNet-101	Cosine Margin	Influence loss on $\mathcal{T}_{\text{old}}$
$\phi_{\text{old}}^{\text{S}}$	-	100%	128	ResNet-101	SoftMax	-
$\phi_{\text{new}-\beta}^{\text{Cos}-\text{S}}$	$\phi_{\text{old}}^{\text{S}}$	100%	128	ResNet-101	Cosine Margin	Influence loss on $\mathcal{T}_{\text{old}}$
$\phi_1^{25\%}$	-	25%	128	ResNet-101	Cosine Margin	-
$\phi_{\text{new}-\beta}^{25\%}$	$\phi_1^{25\%}$	100%	128	ResNet-101	Cosine Margin	Influence loss on $\mathcal{T}_{\text{old }25\%}$
$\phi_1^{90\%}$	-	90%	128	ResNet-101	Cosine Margin	-
$\phi_{\text{new}-\beta}^{90\%}$	$\phi_1^{90\%}$	100%	128	ResNet-101	Cosine Margin	Influence loss on $\mathcal{T}_{\text{old }90\%}$

(a) 'Old Model': the compatible target model for New model. '$\phi_{\text{new}-\beta}^{\text{R152}}$': using ResNet-152 as backbone with the proposed BCT. '$\phi_{\text{new}-\beta}^{\text{R152+256D}}$': using ResNet-152 as backbone and feature dimension of 256 with the proposed BCT. '$\phi_{\text{new}-\beta}^{\text{ReLU}}$': adding a ReLU module after the embedding output of the new model when training with BCT. '$\phi_{\text{old}}^{\text{NS}}$': the old model with normalized SoftMax classifier [3]. '$\phi_{\text{new}-\beta}^{\text{Cos}-\text{NS}}$': the new model with cosine margin classifier [4] and trained by BCT with $\phi_{\text{old}}^{\text{NS}}$ as the old model. '$\phi_{\text{old}}^{\text{S}}$': using standard softmax loss as training loss. '$\phi_{\text{new}-\beta}^{\text{Cos}-\text{S}}$': the new model with cosine margin classifier class [4] and BCT with $\phi_{\text{old}}^{\text{S}}$ as the old model. '$\phi_1^{25\%}$': Model trained by 25% of training data. '$\phi_1^{90\%}$ ': Model trained by 90% of training data.

Comparison Pair	IJB-C 1:1 Verification TAR (%) @ FAR=			
	10^{-5}	10^{-4}	10^{-3}	10^{-2}
$(\phi_{\text{old}}, \phi_{\text{old}})$	59.41	77.86	88.80	95.35
$(\phi_{\text{new}-\beta}^{\text{R152}}, \phi_{\text{old}})$	66.44	80.54	89.87	95.71
$(\phi_{\text{new}-\beta}^{\text{R152+256D}}, \phi_{\text{old}})$	68.45	80.92	89.83	95.84
$(\phi_{\text{new}-\beta}^{\text{ReLU}}, \phi_{\text{old}})$	17.02	34.70	59.82	83.69
$(\phi_{\text{old}}^{\text{S}}, \phi_{\text{old}}^{\text{S}})$	57.30	73.27	85.91	94.45
$(\phi_{\text{new}-\beta}^{\text{Cos}-\text{S}}, \phi_{\text{old}}^{\text{S}})$	48.08	67.11	84.01	94.39
$(\phi_{\text{old}}^{\text{NS}}, \phi_{\text{old}}^{\text{NS}})$	65.56	80.10	89.80	95.61
$(\phi_{\text{new}-\beta}^{\text{Cos}-\text{NS}}, \phi_{\text{old}}^{\text{NS}})$	69.72	81.81	90.39	96.10
$(\phi_1^{25\%}, \phi_1^{25\%})$	29.40	51.77	73.30	88.97
$(\phi_{\text{new}-\beta}^{25\%}, \phi_1^{25\%})$	33.51	52.21	72.78	88.84
$(\phi_1^{90\%}, \phi_1^{90\%})$	74.49	86.58	93.24	97.03
$(\phi_{\text{new}-\beta}^{90\%}, \phi_1^{90\%})$	74.61	86.27	93.19	97.08

(b) Experiments on the IJB-C 1:1 verification task.

Comparison Pair	IJB-C 1:N Retrieval TNIR (%) @ FPIR=			IJB-C 1:N Retrieval Retrieval Rate (%)	
	10^{-3}	10^{-2}	10^{-1}	Rank-1	Rank-5
$(\phi_{\text{old}}, \phi_{\text{old}})$	36.90	59.34	79.18	87.25	92.83
$(\phi_{\text{new}-\beta}^{\text{R152}}, \phi_{\text{old}})$	57.22	68.71	82.58	89.19	94.10
$(\phi_{\text{new}-\beta}^{\text{R152+256D}}, \phi_{\text{old}})$	54.77	69.45	83.60	89.12	94.07
$(\phi_{\text{new}-\beta}^{\text{ReLU}}, \phi_{\text{old}})$	6.98	17.73	33.13	52.58	73.32
$(\phi_{\text{old}}^{\text{S}}, \phi_{\text{old}}^{\text{S}})$	31.53	54.16	73.88	85.42	92.69
$(\phi_{\text{new}-\beta}^{\text{Cos}-\text{S}}, \phi_{\text{old}}^{\text{S}})$	31.44	45.46	69.19	85.37	92.80
$(\phi_{\text{old}}^{\text{NS}}, \phi_{\text{old}}^{\text{NS}})$	40.16	64.32	81.36	89.35	94.39
$(\phi_{\text{new}-\beta}^{\text{Cos}-\text{NS}}, \phi_{\text{old}}^{\text{NS}})$	54.34	71.16	83.40	90.24	94.74
$(\phi_1^{25\%}, \phi_1^{25\%})$	11.34	26.84	54.00	71.67	82.94
$(\phi_{\text{new}-\beta}^{25\%}, \phi_1^{25\%})$	16.48	34.24	57.66	76.89	87.44
$(\phi_1^{90\%}, \phi_1^{90\%})$	57.81	74.52	87.28	91.95	95.72
$(\phi_{\text{new}-\beta}^{90\%}, \phi_1^{90\%})$	61.41	74.57	87.50	91.92	95.51

(c) Experiments on the IJB-C 1:N retrieval task.

Table 2: Robustness analysis of our proposed method against different training factors. When we use the proposed Backward Compatible Training method to train the new model, we change the network structure, feature dimension, data amount and supervision loss, respectively.

New Model	Old Model	Data	Additional Loss
ϕ_{old}	-	50%	-
ϕ^*_{new}	-	100%	-
$\phi_{\text{new}-\ell^2}$	ϕ_{old}	100%	ℓ^2 distance ϕ_{old}
$\phi_{\text{new}-\text{LwF}}$	ϕ_{old}	50%	Learning w/o Forgetting
$\phi_{\text{new}-\beta}$	ϕ_{old}	100%	Influence loss on $\mathcal{T}_{\text{old}}$
$\phi_{\text{new}-\beta-kd}$	ϕ_{old}	100%	Influence loss on $\mathcal{T}_{\text{new}}$
$\phi_{\text{new}-\beta-sys}$	ϕ_{old}	100%	Influence loss on $\mathcal{T}_{\text{new}}$

(a) Training setting for different backward-compatible (new) models. 'Old Model': The compatible target model for New model. '$\phi_{\text{new}-\ell^2}$': The new model with cosine margin classifier [4] and regularized with ℓ^2 distance to ϕ_{old} output feature. '$\phi_{\text{new}-\text{LwF}}$': The new model with cosine margin classifier [4] and adopt Learning w/o Forgetting [1] approach for the new model training. '$\phi_{\text{new}-\beta}$': The new model trained with proposed BCT. '$\phi_{\text{new}-\beta-kd}$': Trained with proposed BCT and use knowledge distillation to bypass obtaining soft supervision labels for the new classes in the new embedding training dataset. '$\phi_{\text{new}-\beta-sys}$': Trained with proposed BCT and use the feature processed by ϕ_{old} as the synthesised classifier for new classes in the growing embedding training dataset.

Comparison Pair	IJB-C 1:1 Verification TAR (%) @ FAR=			
	10^{-5}	10^{-4}	10^{-3}	10^{-2}
$(\phi_{\text{old}}, \phi_{\text{old}})$	59.41	77.86	88.80	95.35
$(\phi^*_{\text{new}}, \phi_{\text{old}})$	0.0	0.0	0.0	0.0
$(\phi_{\text{new}-\ell^2}, \phi_{\text{old}})$	0.5	3.10	10.32	31.98
$(\phi_{\text{new}-\text{LwF}}, \phi_{\text{old}})$	57.09	77.26	88.65	95.46
$(\phi_{\text{new}-\beta}, \phi_{\text{old}})$ (Proposed)	66.06	80.25	89.79	95.62
$(\phi_{\text{new}-\beta-kd}, \phi_{\text{old}})$ (Proposed)	67.82	80.34	89.60	95.74
$(\phi_{\text{new}-\beta-sys}, \phi_{\text{old}})$ (Proposed)	68.35	80.59	89.42	95.23
$(\phi^*_{\text{new}}, \phi^*_{\text{new}})$	76.77	86.96	93.66	97.18

(b) Experiments on the IJB-C 1:1 verification task.

Comparison Pair	IJB-C 1:N Retrieval TNIR (%) @ FPIR=			Retrieval Rate (%)	
	10^{-3}	10^{-2}	10^{-1}	Rank-1	Rank-5
$(\phi_{\text{old}}, \phi_{\text{old}})$	36.90	59.34	79.18	87.25	92.83
$(\phi^*_{\text{new}}, \phi_{\text{old}})$	0.0	0.0	0.0	0.0	0.01
$(\phi_{\text{new}-\ell^2}, \phi_{\text{old}})$	0.14	0.50	2.93	8.41	20.25
$(\phi_{\text{new}-\text{LwF}}, \phi_{\text{old}})$	35.89	59.27	79.00	87.35	93.12
$(\phi_{\text{new}-\beta}, \phi_{\text{old}})$ (Proposed)	52.58	67.23	82.34	88.95	93.95
$(\phi_{\text{new}-\beta-\text{kd}}, \phi_{\text{old}})$ (Proposed)	56.20	69.02	82.50	89.01	93.96
$(\phi_{\text{new}-\beta-\text{sys}}, \phi_{\text{old}})$ (Proposed)	59.48	70.70	82.97	90.09	94.53
$(\phi^*_{\text{new}}, \phi^*_{\text{new}})$	61.93	76.88	87.70	92.11	95.72

(c) Experiments on the IJB-C 1:N search task.

Table 3: We experiment with different approaches towards compatibility of comparison pair. In (a), we illustrate the training details of all the models we trained. In (b), we show the benchmarking results of 1:1 verification on the IJB-C dataset [2]. In (c), we show the benchmarking results of 1:N search on the IJB-C dataset.

New Model	Old Model	Data	Additional Loss
ϕ_1	-	25%	-
ϕ_2	ϕ_1	50%	Influence loss on $\mathcal{T}_1$
ϕ_3	ϕ_2	100%	Influence loss on $\mathcal{T}_2$

(a) 'Old Model': the compatible target model for New model. 'ϕ_1': model trained with 25% of training data. 'ϕ_2': Model trained by 50% of training data and proposed BCT with ϕ_1. 'ϕ_3': Model trained by all of training data and proposed BCT with ϕ_2.

Comparison Pair	IJB-C 1:1 Verification TAR (%) @ FAR=			
	10^{-5}	10^{-4}	10^{-3}	10^{-2}
(ϕ_1, ϕ_1)	29.24	41.45	73.40	89.06
(ϕ_2, ϕ_1)	36.13	56.34	75.14	89.54
(ϕ_3, ϕ_1)	32.14	53.98	71.95	88.69
(ϕ_2, ϕ_2)	58.01	75.96	87.97	95.18
(ϕ_3, ϕ_2)	64.84	80.40	89.84	95.84
(ϕ_3, ϕ_3)	73.61	85.66	92.88	96.90

(b) Experiments on the IJB-C 1:1 verification task.

Comparison Pair	IJB-C 1:N Retrieval TNIR (%) @ FPIR=			Retrieval Rate (%)	
	10^{-3}	10^{-2}	10^{-1}	Rank-1	Rank-5
(ϕ_1, ϕ_1)	11.30	22.57	54.39	71.52	82.87
(ϕ_2, ϕ_1)	19.21	39.00	59.85	78.22	87.81
(ϕ_3, ϕ_1)	14.85	36.10	56.35	77.18	87.48
(ϕ_2, ϕ_2)	38.59	56.07	78.50	86.97	92.49
(ϕ_3, ϕ_2)	50.81	66.09	82.97	89.26	94.11
(ϕ_3, ϕ_3)	59.22	74.12	86.70	91.61	95.53

(c) Experiments on the IJB-C 1:N verification task.

Table 4: Robustness analysis of our proposed method against different training factors. When we use the proposed Backward Compatible Training method to train the new model, we change the network structure, feature dimension, data amount and supervision loss, respectively.

ification. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1041–1049. ACM, 2017. 2

[4] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5265–5274, 2018. 1, 2, 3