Modeling the Background for Incremental Learning in Semantic Segmentation
Supplementary Material

Fabio Cermelli[1,2], Massimiliano Mancini[2,3,4], Samuel Rota Bulò[5], Elisa Ricci[3,6], Barbara Caputo[1,2]
[1]Politecnico di Torino, [2]Italian Institute of Technology, [3]Fondazione Bruno Kessler,
[4]Sapienza University of Rome, [5]Mapillary Research, [6]University of Trento
{fabio.cermelli, barbara.caputo}@polito.it, mancini@diag.uniroma1.it,
samuel@mapillary.com, eliricci@fbk.eu

1. Qualitative results

We provide qualitative results of our method and the baselines on images of the validation sets of ADE20k [9] and Pascal-VOC 2012 [3] for different incremental learning scenarios. The results are shown in the accompanying video (*i.e. video.mp4*), together with an illustration of the background shift problem. As the video shows, the naive strategy of tuning network parameters (FT) for each step independently fails, forgetting all the old classes. Similarly, since the background shift exacerbates catastrophic forgetting, previous incremental learning methods that do not model the background (LwF [5], LwF-MC [7], ILT [6]) are not able to either maintain previous knowledge or learn novel concepts. Our method (MiB) instead can both learn new classes while not forgetting old ones, showing its effectiveness in modeling the background shift for incremental learning in semantic segmentation.

2. How should we use the background?

As highlighted in the main paper, an important design choice for incremental learning in semantic segmentation is how to use the background. In particular, since the background class is present both in old and new classes, it can be considered either in the supervised cross-entropy loss, in the distillation component or in both. For our method and all the baselines (LwF [5], Lwf-MC [7], ILT [6]), we considered the latter case (*i.e.* background in both). However, a natural question arises on how different choices for the background would impact the final results. In this section we investigate this point.

We start from the LwF-MC [7] baseline, since it is composed of multiple binary classifiers and allows to easy decouple modifications on the background from the other classes. We then test two variants:

- **LwF-MC-D** ignores the background in the classification loss, using as target for the background the probability given by $f_{\theta^{t-1}}$.
- **LwF-MC-C** ignores the background in the distillation loss, using only the supervised signal from the ground-truth.

In Table 1 and 2 we report the results of the two variants for the overlapped scenarios of the Pascal VOC dataset and the *50-50* scenario of ADE20K respectively. Together with the two variants, we report the results of our method (MiB), the offline training upper-bound (Joint) and the LwF-MC version employed in the paper which uses the background in both binary cross-entropy and distillation, blending the two components with a hyper-parameter.

As the tables show, the three variants of Lwf-MC exhibit different trade-offs among learning new knowledge and remembering the past one. In particular, LwF-MC-C learns very well new classes, being always the most performing variant on the last incremental step. However, it suffers a significant drop in the old knowledge, showing its inability to tackle the catastrophic forgetting problem.

LwF-MC-D shows the opposite trend. It maintains very well the old knowledge, being the best variant in old classes for every setting. However, it is very intransigent [1] *i.e.* it is not able to correctly learn new classes, thus obtaining the worst performances on them.

As expected, LwF-MC which considers the background in both cross-entropy and distillation achieves a trade-off among learning new knowledge, as in LwF-MC-C, while preserving the old one, as in LwF-MC-D.

As the tables show, our MiB approach models the background more effectively, achieving the best trade-off among learning new knowledge and preserving old concepts. In particular, our method is the best by a margin in all scenarios for the new classes, while for old ones it is either better or comparable to the performance of the intransigent LwF-MC-D method. The only scenarios where it shows lower

Table 1: Comparison of different implementations of LwF-MC on the Pascal-VOC 2012 *overlapped* setup.

	19-1			**15-5**			**15-1**		
Method	*1-19*	*20*	*all*	*1-15*	*16-20*	*all*	*1-15*	*16-20*	*all*
LwF-MC-C	44.6	17.6	43.2	41.6	42.2	41.8	4.4	8.6	5.4
LwF-MC	64.4	13.3	61.9	58.1	35.0	52.3	6.4	8.4	6.9
LwF-MC-D	**71.3**	3.6	**68.0**	73.7	21.0	60.5	**41.1**	3.1	**31.6**
MiB	70.2	**22.1**	67.8	**75.5**	**49.4**	**69.0**	35.1	**13.5**	29.7
Joint	77.4	78.0	77.4	79.1	72.6	77.4	79.1	72.6	77.4

Table 2: Comparison of different implementations of LwF-MC on the *50-50* setting of the ADE20K dataset

Method	*1-50*	*51-100*	*101-150*	*all*
LwF-MC-C	8.0	7.2	19.3	11.5
LwF-MC	27.8	7.0	10.4	15.1
LwF-MC-D	**39.1**	10.9	6.7	18.7
MiB	35.5	**22.2**	**23.6**	**27.0**
Joint	51.1	38.3	28.2	38.9

performances are the multi-step ones. Indeed in these scenarios, the multiple learning episodes make preserving old knowledge harder, and an intransigent method is less prone to forgetting since it is biased to old classes. However, the intransigence is not the right solution if the number of old and new classes are balanced, as in the *50-50* scenario of ADE20k, since the overall performances will be damaged.

3. Per class results on Pascal-VOC 2012

From Table 3 to 8, we report the results for all classes of the Pascal-VOC 2012 dataset. As the tables show, MiB achieves the best results in the majority of classes (i.e. at least 14/20 in the 19-1 scenarios, 13/20 in the 15-5 and 16/20 in the 15-1 ones) being either the second best or comparable to the top two in all the others. Remarkable cases are the ones where we learn classes that are either similar in appearance (*e.g.* bus and train) or appear in similar contexts (*e.g.* sheep and cow): for those pairs, our model outperforms the competitors by a margin in both old classes (*i.e.* bus and cow in the 15-5 and 15-1 scenarios) and new ones (*i.e.* sheep and train). These results show the capability of MiB to not only learn new knowledge while preserving the old one, but also to learn discriminative features for difficult cases during different learning steps.

4. Validation protocol and hyper-parameters

In this work, we follow the protocol of [2] for setting the hyper-parameters in continual learning. The protocol works in three steps and does not require *any* data of old tasks. First, we split the training set of the current learning step into train and validation sets. We use 80% of the data for training and 20% for validation. Note that the validation set contains only labels for the current learning step.

Second, we set general hyper-parameters values (*e.g.* learning rate) as the ones achieving the highest accuracy in the new set of classes with the fine-tuned model. Since we tested multiple methods, we wanted to ensure fairness in terms of hyper-parameters used, without producing biased results. To this extent, this step is held out only once starting from the fine-tuned model and fixing the hyper-parameters for all the methods. In particular, we set the learning rate as 10^{-3} for the incremental steps in all datasets and settings.

As a final step, we set the hyper-parameters specific of the continual learning method as the highest values (to ensure minimum forgetting) with a tolerated decay on the performance on the new classes with respect to the ones achieved by the fine-tuned model (to ensure maximum learning). We set the tolerated decay as 20% of the original performances, exploring hyper-parameters values of the form $A \cdot 10^B$, with $A \in \{1, 5\}$ and $B \in \{-3, \dots, 3\}$. We perform this validation procedure in the first learning step of each scenario, keeping the hyper-parameters fixed for the subsequent ones. Since this procedure is costly, we perform it only for the Pascal-VOC dataset, keeping the hyper-parameters for the large-scale ADE20k. As a result, for the prior focused methods, we obtain a weight of 500 for EWC [4] and PI [8] and 100 for RW [1] in all scenarios. For the data-focused methods we obtain a weight of 100 for the distillation loss of LwF [5], 10 for the one in LwF-MC [7] and 100 for both distillation losses in ILT [6], in all settings. For our MiB method, we obtain a distillation loss weight of 10 for all scenarios except for the *15-1* in Pascal VOC, where the weight is set to 100.

5. Code

Together with the supplementary material, we provide the code used to obtain our results. The code can be found at `https://github.com/fcdl94/MiB`. We implement all the methods using the PyTorch[1] framework, performing our experiments on two NVIDIA Titan RTX GPUs. We provided all the instructions to set up all the incremental learning scenarios of both ADE20k and Pascal VOC, as well as all the requirements for training and testing our method and all the baselines. The instructions are available in the *README.md* file.

References

[1] Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *ECCV*, 2018. 1, 2, 3, 4

[2] Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Gregory Slabaugh, and Tinne

[1] `https://pytorch.org/`

Table 3: Per Class Mean IoU on 19-1 setting of Pascal-VOC 2012. disjoint setup

Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	persn	plant	sheep	sofa	train	tv	**1-19**	**all**
FT	11.9	2.1	1.1	11.6	4.8	6.9	13.5	0.2	0.0	3.8	14.4	0.5	1.5	4.7	0.0	15.8	2.8	1.8	13.5	12.3	5.8	6.2
PI [8]	22.3	1.9	3.4	4.9	2.1	10.6	8.5	0.1	0.1	3.1	12.8	0.2	3.8	4.6	0.0	10.0	5.0	1.1	8.5	14.1	5.4	5.9
EWC [4]	50.7	7.7	21.0	24.1	21.8	35.8	43.9	11.6	2.0	27.0	21.1	23.0	18.7	19.4	1.5	27.8	41.5	5.6	37.4	16.0	23.2	22.9
RW [1]	45.8	5.3	15.1	22.8	17.8	28.9	40.9	7.5	1.3	22.4	20.3	14.5	13.7	16.3	0.8	25.3	31.8	4.8	33.3	15.7	19.4	19.2
LwF [5]	28.1	40.5	53.1	38.8	47.4	46.4	63.6	83.5	35.8	60.1	48.8	76.5	65.3	67.1	83.2	50.2	61.2	42.5	14.2	9.1	53.0	50.8
LwF-MC [7]	79.4	**41.3**	75.6	47.9	51.0	69.6	75.4	78.5	35.1	66.6	49.0	72.7	73.8	71.6	**84.9**	**57.5**	67.7	42.7	56.8	13.2	63.0	60.5
ILT [6]	**83.7**	40.8	80.8	**59.1**	58.4	77.6	**82.4**	82.3	38.9	81.7	50.8	84.8	**86.6**	**81.0**	83.3	56.4	82.2	43.8	57.5	16.4	69.1	66.4
MiB	78.0	40.5	**85.7**	51.6	**64.4**	**79.1**	77.8	**89.9**	**39.2**	**82.3**	**55.4**	**86.2**	82.7	72.2	83.6	56.6	**86.2**	**45.1**	**65.0**	**25.6**	**69.6**	**67.4**
Joint	90.2	42.2	89.5	69.1	82.3	92.5	90.0	94.2	39.2	87.6	56.4	91.2	86.8	88.0	86.8	62.3	88.4	49.5	85.0	78.0	77.4	77.4

Table 4: Per Class Mean IoU on 19-1 setting of Pascal-VOC 2012. overlapped setup

Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	persn	plant	sheep	sofa	train	tv	**1-19**	**all**
FT	23.7	1.9	1.5	9.3	6.9	16.9	8.5	0.0	0.0	9.5	5.3	0.1	2.9	8.8	0.0	15.1	1.0	0.7	16.0	12.9	6.8	7.1
PI [8]	33.1	4.1	3.6	10.5	8.4	14.7	13.3	0.0	0.1	2.4	4.7	0.1	3.3	7.9	0.0	14.7	0.8	2.7	17.8	14.0	7.5	7.8
EWC [4]	60.7	14.8	21.2	33.8	36.9	54.4	45.6	2.6	1.4	33.0	13.3	19.1	23.8	39.2	2.2	34.6	21.8	6.4	47.1	14.0	26.9	26.3
RW [1]	57.5	12.1	15.4	29.6	32.9	50.7	40.0	1.3	0.8	30.7	10.7	12.6	18.6	32.9	0.8	30.7	17.5	5.5	42.7	14.2	23.3	22.9
LwF [5]	36.6	35.1	62.0	32.9	47.5	31.6	51.5	77.9	36.5	67.7	44.3	71.4	68.6	66.2	82.2	49.6	58.7	41.1	11.9	8.5	51.2	49.1
LwF-MC [7]	67.2	37.9	77.8	40.6	57.0	54.5	77.4	88.4	37.2	76.8	49.1	83.4	82.3	71.0	**85.2**	55.6	81.9	**46.0**	54.9	13.3	64.4	61.9
ILT [6]	**87.2**	**39.0**	80.6	**53.5**	57.0	80.3	76.0	74.3	37.6	81.1	44.6	83.1	**84.4**	81.6	82.4	54.5	82.7	38.9	56.1	12.3	67.1	64.4
MiB	78.1	36.2	**86.8**	49.4	**72.7**	**80.8**	**78.2**	**90.8**	**38.3**	**82.0**	**51.9**	**86.7**	82.8	**76.9**	83.8	**58.8**	**84.4**	45.7	**68.5**	**22.1**	**70.2**	**67.8**
Joint	90.2	42.2	89.5	69.1	82.3	92.5	90.0	94.2	39.2	87.6	56.4	91.2	86.8	88.0	86.8	62.3	88.4	49.5	85.0	78.0	77.4	77.4

Tuytelaars. Continual learning: A comparative study on how to defy forgetting in classification tasks. 2019. 2

[3] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 1

[4] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017. 2, 3, 4

[5] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE T-PAMI*, 40(12):2935–2947, 2017. 1, 2, 3, 4

[6] Umberto Michieli and Pietro Zanuttigh. Incremental learning techniques for semantic segmentation. In *ICCV-WS*, pages 0–0, 2019. 1, 2, 3, 4

[7] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *CVPR*, 2017. 1, 2, 3, 4

[8] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *ICML*, 2017. 2, 3, 4

[9] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, 2017. 1

Table 5: Per Class Mean IoU on 15-5 setting of Pascal-VOC 2012. disjoint setup

Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	persn	plant	sheep	sofa	train	tv	**1-15**	**16-20**	**all**
FT	6.1	0.0	0.2	8.3	0.1	0.0	0.1	0.0	0.0	0.0	0.0	0.0	1.8	0.0	0.0	24.6	24.3	36.2	32.5	50.2	1.1	33.6	9.2
PI [8]	8.8	0.0	0.2	10.5	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0	0.4	0.0	0.0	25.6	24.7	34.3	34.1	52.0	1.3	34.1	9.5
EWC [4]	58.8	4.1	56.4	46.2	44.4	4.3	67.4	3.6	2.3	14.8	10.3	12.4	51.6	20.4	2.9	28.8	32.2	35.6	35.5	56.3	26.7	37.7	29.4
RW [1]	51.1	1.5	36.9	42.9	27.5	2.1	47.4	1.1	1.2	6.1	5.3	3.1	31.2	10.5	1.0	27.7	29.8	35.7	34.7	**56.6**	17.9	36.9	22.7
LwF [5]	63.1	40.1	72.4	52.1	67.0	6.7	80.3	84.2	31.1	5.7	51.3	82.0	75.0	79.4	85.6	35.3	27.1	37.0	37.0	50.5	58.4	37.4	53.1
LwF-MC [7]	78.1	**42.3**	78.9	62.1	78.6	47.3	84.6	89.1	35.0	26.2	50.5	86.6	77.6	**84.9**	**86.0**	35.0	35.2	**40.8**	49.2	45.9	67.2	41.2	60.7
ILT [6]	79.4	42.0	80.5	63.9	**80.4**	12.8	**86.0**	90.2	30.7	6.7	**53.3**	83.2	73.0	80.7	85.0	**36.9**	29.9	36.8	38.3	55.7	63.2	39.5	57.3
MiB	**84.4**	39.4	**87.5**	**65.2**	77.8	**61.0**	**86.0**	**90.9**	**35.3**	**60.3**	53.0	**88.2**	**80.4**	82.4	85.3	28.7	**46.0**	34.7	**54.4**	52.7	**71.8**	**43.3**	**64.7**
Joint	90.2	42.2	89.5	69.1	82.3	92.5	90.0	94.2	39.2	87.6	56.4	91.2	86.8	88.0	86.8	62.3	88.4	49.5	85.0	78.0	79.1	72.6	77.4

Table 6: Per Class Mean IoU on 15-5 setting of Pascal-VOC 2012. overlapped setup

Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	persn	plant	sheep	sofa	train	tv	**1-15**	**16-20**	**all**
FT	13.4	0.1	0.0	15.6	0.8	0.0	0.3	0.0	0.0	0.0	0.0	0.0	0.9	0.0	0.0	30.9	21.6	32.8	34.9	45.1	2.1	33.1	9.8
PI [8]	7.8	0.0	0.0	12.9	0.3	0.0	0.3	0.0	0.0	0.0	0.0	0.0	2.7	0.0	0.0	33.2	22.2	33.2	36.1	42.0	1.6	33.3	9.5
EWC [4]	67.3	12.8	50.5	52.9	35.0	24.7	41.7	1.2	1.0	9.8	5.7	3.7	42.9	15.4	0.6	31.8	26.3	32.1	42.0	45.0	24.3	35.5	27.1
RW [1]	61.2	6.7	33.8	48.1	24.4	9.3	22.3	0.3	0.5	3.5	0.2	1.1	31.8	6.4	0.1	32.1	25.8	31.9	38.7	45.9	16.6	34.9	21.2
LwF [5]	64.5	40.2	72.8	56.9	57.3	9.5	82.6	88.6	33.2	8.9	48.4	81.9	75.0	78.2	84.9	34.7	27.8	33.1	39.6	48.0	58.9	36.6	53.3
LwF-MC [7]	60.6	38.9	74.7	41.6	67.2	10.8	81.4	88.8	**38.7**	4.3	47.4	82.2	69.9	78.9	**85.8**	28.4	28.5	**34.1**	36.4	47.8	58.1	35.0	52.3
ILT [6]	77.4	**40.3**	78.9	61.9	78.7	53.5	86.1	88.7	33.8	15.9	51.1	83.2	80.2	79.8	85.0	**39.5**	30.9	31.0	49.3	52.6	66.3	40.6	59.9
MiB	**86.6**	39.3	**88.9**	**66.1**	**80.8**	**86.6**	**90.1**	**92.5**	38.0	**64.6**	**56.4**	**89.6**	**80.5**	**86.5**	85.7	30.2	**52.9**	31.3	**73.2**	**59.5**	**75.5**	**49.4**	**69.0**
Joint	90.2	42.2	89.5	69.1	82.3	92.5	90.0	94.2	39.2	87.6	56.4	91.2	86.8	88.0	86.8	62.3	88.4	49.5	85.0	78.0	79.1	72.6	77.4

Table 7: Per Class Mean IoU on 15-1 setting of Pascal-VOC 2012. disjoint setup

Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	persn	plant	sheep	sofa	train	tv	**1-15**	**16-20**	**all**
FT	0.3	0.0	0.0	2.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	8.8	0.2	1.8	0.6
PI [8]	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.3	8.6	0.0	1.8	0.4
EWC [4]	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	3.6	0.0	0.0	0.0	0.0	0.0	0.0	7.3	7.0	7.4	0.3	4.3	1.3
RW [1]	0.0	0.0	0.0	0.2	0.0	0.0	0.0	0.0	0.0	0.0	2.2	0.0	0.0	0.0	0.0	0.0	0.0	8.1	10.5	8.2	0.2	5.4	1.5
LwF [5]	0.0	0.0	0.0	0.0	0.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	10.7	0.0	0.0	1.9	8.2	7.9	0.8	3.6	1.5
LwF-MC [7]	0.0	6.3	0.8	0.0	1.1	0.0	0.1	0.3	0.0	0.0	0.0	0.0	0.2	0.0	59.0	0.0	9.5	2.9	11.9	**11.0**	4.5	7.0	5.2
ILT [6]	3.7	0.0	2.9	0.0	12.8	0.0	0.0	0.1	0.0	0.0	**21.2**	0.1	0.4	0.6	13.6	0.0	0.0	11.6	8.3	8.5	3.7	5.7	4.2
MiB	**53.6**	**38.9**	**53.6**	**17.7**	**62.7**	**36.5**	**71.2**	**60.1**	**1.1**	**35.2**	8.1	**57.6**	**55.0**	**62.1**	**79.4**	**10.2**	**14.2**	**11.9**	**18.2**	10.1	**46.2**	**12.9**	**37.9**
Joint	90.2	42.2	89.5	69.1	82.3	92.5	90.0	94.2	39.2	87.6	56.4	91.2	86.8	88.0	86.8	62.3	88.4	49.5	85.0	78.0	79.1	72.6	77.4

Table 8: Per Class Mean IoU on 15-1 setting of Pascal-VOC 2012. overlapped setup

Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	persn	plant	sheep	sofa	train	tv	**1-15**	**16-20**	**all**
FT	2.6	0.0	0.0	0.7	0.0	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	9.2	0.2	1.8	0.6
PI [8]	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	9.1	0.0	1.8	0.5
EWC [4]	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	3.6	0.0	0.0	0.0	0.0	0.0	0.0	7.3	7.0	7.4	0.3	4.3	1.3
RW [1]	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	8.7	11.2	6.3	0.0	5.2	1.3
LwF [5]	3.7	0.1	0.0	2.5	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.0	9.0	0.0	0.0	1.6	8.9	8.8	1.0	3.9	1.8
LwF-MC [7]	0.0	7.2	5.2	0.0	25.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.2	1.3	56.2	0.0	4.9	0.2	8.6	**28.2**	6.4	8.4	6.9
ILT [6]	20.0	0.0	3.2	6.3	2.3	0.0	0.0	0.0	**0.3**	5.1	**19.0**	0.0	9.1	0.0	8.7	0.0	0.0	**21.0**	9.9	8.1	4.9	7.8	5.7
MiB	**31.3**	**25.4**	**26.7**	**26.9**	**46.1**	**31.0**	**63.6**	**52.8**	0.1	**11.0**	9.4	**52.4**	**41.2**	**28.1**	**80.7**	**17.6**	**13.1**	15.3	**15.3**	6.2	**35.1**	**13.5**	**29.7**
Joint	90.2	42.2	89.5	69.1	82.3	92.5	90.0	94.2	39.2	87.6	56.4	91.2	86.8	88.0	86.8	62.3	88.4	49.5	85.0	78.0	79.1	72.6	77.4