Multi-Domain Semantic Segmentation via the Principle of Rate Reduction

Domas Buracas
UC Berkeley
dominykas@berkeley.edu

Nathan Miller
UC Berkeley
nathan_miller23@berkeley.edu

Abstract

We seek to further validate the objective and theoretical principles introduced in Maximal Coding Rate Reduction (MCR^2) [1] by scaling it to the task of semantic segmentation in real world environments, where the number of labels per inference and size of datasets is significantly greater than the classification datasets STL10 and CIFAR100. Additionally, we show that applying a learned mask to filter high-frequency background pixels can greatly improve both the empirical and qualitative performance of MCR^2. Finally, while the MCR2 objective does not appear to improve in learning domain agnostic features to assist with domain generalization, it improves upon standard supervised learning losses when the input space contains samples from many varied domains.

1 Introduction

The problem of image segmentation has interesting applications in a diverse array of fields such as biomedical imaging [13] and self-driving cars [2]. The recent proliferation of deep learning and the widespread use of CNNS has greatly improved the state of the art in image segmentation [2]. However, despite the expressive power of these deep learning models, state of the art architectures still fail to learn across diverse domains and to generalize to similar but previously unseen domains. [8][9]. Thus, we hope to leverage the power of modern deep neural architectures with the principled theory and intuition of robust latent space representations outlined in MCR^2 [1].

By encouraging the learning of features that are maximally diverse, the MCR^2 objective will result in class-specific features that are sufficient for classification in multiple domains. This diversity, quantified by the discriminative loss, is maximized by the MCR^2 objective, and provides theoretical justification for the objective's application to the multiple domain semantic segmentation setting.

Finally, MCR^2 conatins as a component of its object a compression loss, which indicates how correlated the intra-class representations are. While certain aspects of natural images are difficult to compress, we show that, when the semantically ineffectual information is removed, MCR^2 excels at learning expressive features of semantic instances.

1.1 Semantic Segmentation

Image segmentation can be thought of as a pixel-wise classification of the image. In the more general case of semantic segmentation, each pixel is assigned a semantic class, whereas in instance segmentation, each pixel is classified based on the object it belongs to in the image [2]. The first semantic segmentation milestone occurred with the advent of the fully convolutional net, which adopted popular CNN approaches by replacing all fully connected layers with convolutional layers to more efficiently make pixel-wise predictions from context patches [12]. Building off of this idea, the popular U-Net and V-net architectures forgo pixel-by-pixel prediction and instead decode the entire image in one forward pass through transpose convolutional upsampling [13][14]. Other approaches

Preprint. Under review.

use auto-regressive techniques such as RNNs [15] or LSTMs [16] in an attempt to encode both local and global information. The High Resolution Net (HRNet) builds off of this intuition by constructing both deep, low resolution and shallow, high resolution feature maps in parallel and exchanging information between resolutions [17]. Finally, other models attempt to leverage attention maps in order to more efficiently encode and decode the image [18][19].

Thus, semantic segmentation is clearly a rapidly changing, very active field. Rather than focus on the state of the art, we instead choose to apply the MCR^2 objective to the well known U-net. The reason for doing so is twofold. First, we compare to a cross entropy baseline, trained on the same U-net backbone. Thus, we believe the comparison of the results we achieved will translate proportionally to newer methods, which also tend to rely on cross entropy in some capacity. Additionally, the U-net architecture maintains the spatial dimension of the input throughout, allowing for intuitive application of the pixel-wise labels to the learned feature vectors.

1.2 Domain Generalization

Domain generalization is a particular form of transfer learning that attempts to solve the problem of leveraging expertise acquired in a source domain to improve learning in a similar target domain [4]. Domain generalization in particular assumes that the target domain is unknown and unavailable until test time, and thus the model must learn entirely form the source domains [5].

Common approaches to domain generalization can be divided into two categories: those that attempt to learn robust representations that are invariant to domain shift [5][6][8][9], or those that learn invariant models directly [7][10][11].

The motivating assumptions that underpin the former category are that a good domain generalization representation minimizes variation across source domains and retains the conditional structure between the input features and the labels [5]. Thus, one approach is to find a latent space encoded by a kernel function $k : X \times X \longrightarrow \mathbb{R}$, with implicit feature map $x \mapsto \phi(x)$ such that $X \perp Y | \phi(X)$, where the invariance of the conditional structure of X and Y is encoded by the conditional independence given the latent representation [6]. Other approaches seek to find domain invariant representations using adversarial autoencoders [8], or meta-learning [9]. While these methods produce good accuracy results, they are difficult and time-consuming to train.

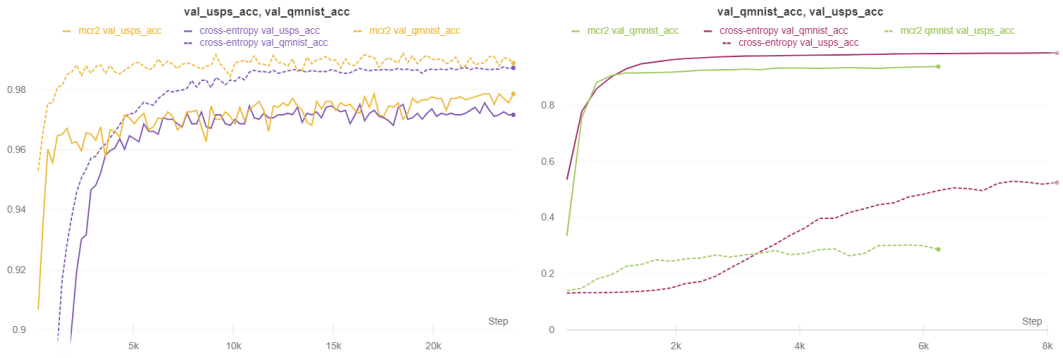
The second category of approaches focuses on learning domain-invariant models directly. For example, Xu et.al. trains a domain-specific classifier for each domain, while imposing a low rank constraint on their cross-domain likelihood predictions [7]. The motivating assumption underpinning this category of approaches is that the input space can be divided into disjoint sets of domain specific and domain invariant features [10][11]. This assumption was exploited by Li et. al by training a model whose weights at every layer were conditioned on a domain-encoding latent vector for D domains of dimension $D+1$ the last element, viewed as the bias bit, was always on, the ith dimension was 1 for domain i and zero for all other entries [11]. This idea was improved upon by Piratla et al by assuming that the domain specific and domain agnostic portions of parameter space are orthogonal and by only conditioning on domain in the final layer in order to improve model efficiency and offer better theoretical guarantees [12].

2 Experiments

We conduct two series of experiments. In the first, we study the domain generalization and fitting properties of MCR^2. In the second, we scale MCR^2 from a classification loss to semantic segmentation over MNIST-scale images with complex backgrounds mimicking full scale datasets like BDD100k.

We use the following datasets:

- **MNIST** Standard handwritten digits classification dataset (QMNIST)
- **USPS** Handwritten digits classification dataset with different style from MNIST
- **DIGITS** Interleaving of the MNIST and USPS datasets, with multiple epochs of USPS appearing per MNIST epoch due to mnist being ~7x larger
- **MNIST_CIFAR_BG** MNIST instances superimposed on natural-image backgrounds collected from the CIFAR100 dataset



(a) Multi-Domain Accuracy

(b) Cross-Domain Accuracy

Figure 1: Comparison of Cross Entropy and MCR^2 accuracy in multi-domain and cross-domain settings

2.1 Classification Experiments

Using a Resnet18 backbone for feature prediction, we train four models with permutations of cross-entropy or MCR^2 losses and with the MNIST or DIGITS datasets. We then evaluate all the models on held-out validation sets from MNIST and USPS. These two datasets have written digits with different distributions of handwriting, sizing, and post-processing artifacts, making them compellingly different domains.

As seen in Figure [1b], MCR^2 has worse accuracy over both validation sets, supporting the claim that it does not generalize across domains any better than cross-entropy.

In the other plot, Figure [1a], where both models are trained on the complete DIGITS dataset containing both MNIST and USPS, MCR^2 outperforms cross-entropy on both validation sets. Therefore it is able to fit to these two domains better than the baseline.

(a) Validation input from standard MNIST dataset and output predictions

(b) Validation input from MNIST_CIFAR_BG dataset and output predictions

Figure 2: Qualitative performance comparison of pixel-wise semantic segmentation trained with MCR^2 objective and Unet background.

Figure 3: Comparison of MCR^2 loss evolution during training on both MNIST (purple) and MNIST_CIFAR_BG (gray) datasets

Figure 4: (left) Input to the background classifier (right) result of pairwise product of the logits of the background classifier and the input image. The background classifier achieves >99% accuracy after 5 epochs

2.2 Semantic Segmentation Experiments

Next, we apply the MCR^2 objective to the task of semantic segmentation. For the following experiments, we use a U-net that maps an input of shape (W, H, C) to $(W, H, latent\_dim)$. As a baseline, we train an MLP that maps each vector of length $(latent\_dim,)$ to logits across classes, and use cross entropy to provide a loss signal. Additionally, we apply the MCR^2 objective pixel-wise to learn a more discriminative latent space representation. We then use a nearest subspace classifier for the final class prediction in calculating validation loss.

2.2.1 Learned Background Masking

While the validation performance of the MCR2 objective with Unet backbone on MNIST quickly matched that of cross entropy, this obfuscates the fact that, in natural images, the background tends to be the noisiest, highest frequency part of the image. Thus, we train the same model on the MNIST_CIFAR_BG dataset to simulate high frequency backgrounds.

As seen in Figure [2], the qualitative performance on MNIST_CIFAR_BG dataset is far worse. While the model is able to distinguish background and instance pixels, it struggles to accurately classify the instance pixels correctly. A simple explanation for this notable drop in performance is that backgrounds are difficult to compress, and thus the MCR^2 objective struggles to learn meaningful representations of them. This suspicion is confirmed in Figure [3]. While the discriminate loss is equivalent for both models, the representation learned with CIFAR backgrounds has far higher compression loss. We thus see an important dichotomy regarding the background pixels: they are easy to classify, but hard to represent.

To combat this issue, we introduce a separate background classification loss head to our model. The latent feature vector for each pixel is extended by 2, and these last two dimensions, interpreted as binary classification logits, are fed to a cross-entropy loss along with the true classification labels. After the background classifier achieves reached 99% validation accuracy, the train-time predictions of the background classifier are used as a mask to remove high-frequency background pixels before they are fed to MCR^2. Figure 4 shows an example input training image, and the result of pairwise multiplying this learned mask. The background classifier is able to hit this 99% threshhold withing 5 epochs, and thus reduces to problem to one similar to the vanilla MNIST dataset, where all background pixels were uniform.

As Seen in Figure [5] and [6], both the qualitative and quantitative results of the model are improved by background masking. Not only is the model able to more accurately classify instance pixels, as well as the pixels on the boundary of the instance, but we see a more favorable loss evolution as well. The background masking allows for the learning of a more within-class compressible feature space. Additionally, the discrimative loss is also higher, as the diversity of the feature space is no longer dominated by the high-frequency background pixels.

Finally, we compare both the masked and non-masked MCR^2 models to a cross entropy baseline. As seen in Figure [5], the MCR^2 objective with background masking outperforms both the non-masked MCR^2 and cross entropy models.

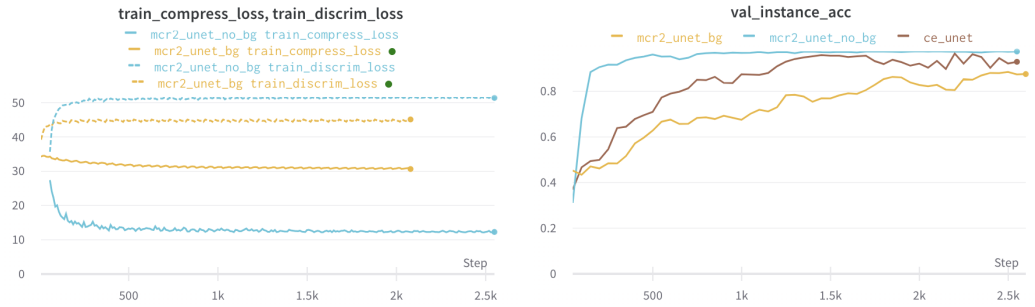


Figure 5: Comparison of quantitative performance impact of background masking. (left) MCR^2 loss evolution for model with (blue) and without (yellow) background masking (right) Validation prediction accuracy for non-background pixels by MCR^2 w/ background masking (blue) w/o background masking (yellow) and cross entropy baseline (brown)

??

Figure 6: Comparison of qualitative performance impact of background masking. (left) Input and semantic segmentation prediction for model trained without background masking (right) Input and semantic segmentation prediction for model trained with background masking

2.2.2 Feature Space Analysis

To better understand the effect of background masking on semantic segmentation performance, we analyze the effect it has on the learned representations. In Figure [7], we compare the cosine similarity of the learned feature vectors for 25,000 validation pixels both with and without background masking. As seen in the figure, the features learned without background masking have more cross-correlation. Several of the off-block-diagonal entries have high cosine similarity, which further confirms the the fact that MCR2 without background masking fails to discrimitate between instance classes. Additionall, the top left corner of consine similarity matrix without background masking shows the intra-class correlation of the background class. As can be seen, the background class consine similarities are sparse compared to other classes, a result of it being hard to compress.

On the other hand, the cosine similarities of the features learned with background masking exhibit the properties we desire. Off-block-diagonal cosine similarities are all close to zero, indicating adequate inter-class discrimination. Additionally, the block diagonal entries are closer to 1, indicating better within class correlation.

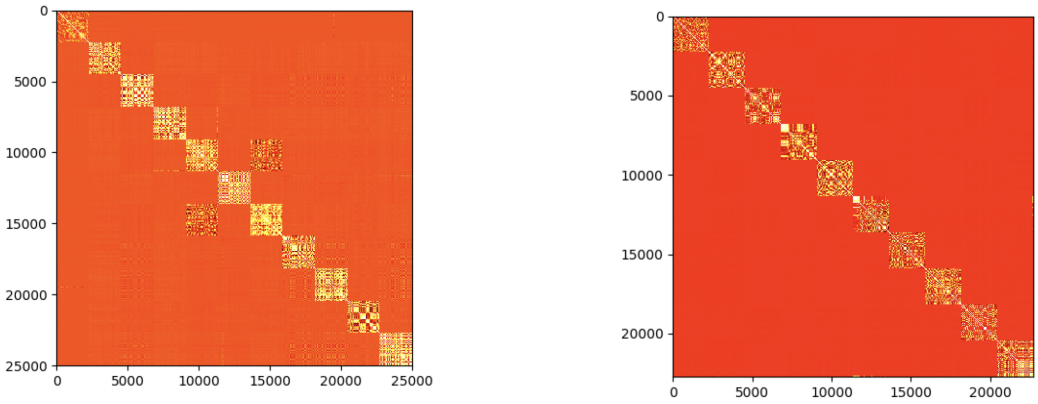


Figure 7: Comparison of cosine similarity across validation instances for MCR^2 trained w/o background masking (left) and w/ background masking (right). 25 thousand validation pixels are randomly sampled and sorted by label. Note that the right figure does not include instances from the background class, as no MCR^2 representation was learned for the background in this case.

3 Conclusion and Future Work

We have shown that MCR^2 can learn on multiple domains better than cross-entropy, but that it does not necessarily generalize to unseen domains. We have also extended it's original classification formulation to semantic segmentation where it outperforms cross-entropy on our simple task after factoring out distracting, high-frequency background information.

As indicated by the success of background masking, some pixels are more important to the semantic classification task, and the learned latent representation, than others. The background masking was an intuitive and naive first step in this direction. Additional improvement could be obtained by identifying instance boundary pixels and learning to only classify those. Additionally, the pixel-wise application of the labels results in superfluously high-resolution classification. Adaptivley down-sampling the labels, and applying the MCR^2 objective at a lower spatial resolution, would enable faster computation. Such downsampling would be essential to scale the method to meaningful real-world semantic segmentation datasets such as BDD100K.

Another direction of interest would be identifying and learning subspaces encoding differences between domains, and factoring them out of currently learned class-prediction subspaces.

References

[1] Yaodong Yu, Kwan Ho Ryan Chan, Chong You, Chaobing Song and Yi Ma. Learning Diverse and Discriminative Representations via the Principle of Maximal Coding Rate Reduction, 2020; arXiv:2006.08558.

[2] Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz and Demetri Terzopoulos. Image Segmentation Using Deep Learning: A Survey, 2020; arXiv:2001.05566.

[3] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan and Trevor Darrell. BDD100K: A Diverse Driving Dataset for Heterogeneous Multitask Learning, 2018; arXiv:1805.04687.

[4] S. J. Pan and Q. Yang. A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering, 22(10):1345–1359, October 2010a

[5] Krikamol Muandet, David Balduzzi and Bernhard Schölkopf. Domain Generalization via Invariant Feature Representation, 2013; arXiv:1301.2115.

[6] Ya Li, Mingming Gong, Xinmei Tian, Tongliang Liu and Dacheng Tao. Domain Generalization via Conditional Invariant Representation, 2018; arXiv:1807.08479.

[7] Xu Z., Li W., Niu L., Xu D. Exploiting Low-Rank Structure from Latent Domains for Domain Generalization, 2014 In: Fleet D., Pajdla T., Schiele B., Tuytelaars T. (eds) Computer Vision – ECCV 2014. ECCV 2014. Lecture Notes in Computer Science, vol 8691. Springer, Cham. https://doi.org/10.1007/978-3-319-10578-9_41

[8] H. Li, S. Jialin Pan, S. Wang, and A. C. Kot. Domain generalization with adversarial feature learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 5400–5409, 2018.

[9] Keyu Chen, Di Zhuang and J. Morris Chang. Discriminative Adversarial Domain Generalization with Meta-learning based Cross-domain Validation, 2020; arXiv:2011.00444.

[10] Da Li, Yongxin Yang, Yi-Zhe Song and Timothy M. Hospedales. Deeper, Broader and Artier Domain Generalization, 2017; arXiv:1710.03077.

[11] Vihari Piratla, Praneeth Netrapalli and Sunita Sarawagi. Efficient Domain Generalization via Common-Specific Low-Rank Decomposition, 2020; arXiv:2003.12815.

[12] Jonathan Long, Evan Shelhamer and Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation, 2014; arXiv:1411.4038.

[13] Olaf Ronneberger, Philipp Fischer and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation, 2015; arXiv:1505.04597.

[14] Fausto Milletari, Nassir Navab and Seyed-Ahmad Ahmadi. V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation, 2016; arXiv:1606.04797.

[15] Francesco Visin, Marco Ciccone, Adriana Romero, Kyle Kastner, Kyunghyun Cho, Yoshua Bengio, Matteo Matteucci and Aaron Courville. ReSeg: A Recurrent Neural Network-based Model for Semantic Segmentation, 2015; arXiv:1511.07053.

[16] W. Byeon, T. M. Breuel, F. Raue, and M. Liwicki, "Scene labeling with lstm recurrent neural networks," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3547–3555.

[17] Liang-Chieh Chen, Yi Yang, Jiang Wang, Wei Xu and Alan L. Yuille. Attention to Scale: Scale-aware Semantic Image Segmentation, 2015; arXiv:1511.03339.

[18] Hanchao Li, Pengfei Xiong, Jie An and Lingxue Wang. Pyramid Attention Network for Semantic Segmentation, 2018; arXiv:1805.10180.