Multi-task Curriculum Learning for Computer Vision
CS 381V Project Report

Santhosh K. Ramakrishnan
The University of Texas at Austin
srama@cs.utexas.edu

Wonjoon Goo
The University of Texas at Austin
wonjoon@cs.utexas.edu

Abstract

Representation Learning has been one of the fundamental topics in Computer Vision over the past few decades. Learning representations from multiple tasks has been pursued actively in the recent deep learning and computer vision literature. Multi-Task Learning and Curriculum Learning have been explored in various degrees in the Machine Learning and Computer Vision literatures. In this project, we explore the importance of sampling the data through some form of a curriculum within the context of Multi-Task Learning. We explore various problems in Computer Vision using Multi-Task Curriculum Learning and provide some insights into it's efficancy and applicability under these different domains.

1. Introduction

Representation learning is a fundamental topic that has been extensively explored in the vision literature. Handcrafted features [14, 25, 3] have been used for successfully for several decades. These were subsequently replaced (to a good extent) by learnt representations [17, 33, 34, 15] aka Deep Learning. The primary success of deep learning has been the availability of large labelled datasets and computation power. The current trend is to pre-train a network on ImageNet and use the learnt representations for subsequent tasks. However, while ImageNet has been vital for development in Computer Vision, the ImageNet object classification task may not be an appropriate pre-training mechanism for all tasks. Firstly, it applies only to images from real domain (not cartoons, sketches, segmented maps, etc). Secondly, it mainly captures semantic knowledge and not geometric knowledge. Finally, collecting a dataset like ImageNet is very expensive and such a strong ontology may not be defined for all tasks.

Since ImageNet cannot be used to learn features for all possible visual tasks, other alternatives have been explored. The infeasibility to collect large labelled datasets such as ImageNet has led to the rise of unsupervised / self-supervised learning strategies [7, 16, 30, 36]. While the data available for unsupervised tasks are massive, none of the tasks proposed have succeeded in replacing ImageNet pre-training, indicating the importance of human labelling.

Most of the works in the recent literature have explored using single tasks for visual representation learning (supervised / unsupervised). There has been limited work exploring the role of multi-task learning for learning visual representations [35, 8]. The human vision system has developed by performing multiple tasks (supervised and unsupervised) over extended periods of time. Another aspect of the human vision system is the role of learning each of the tasks at various difficulty levels. This has been explored in the cognitive science [10], machine learning [4, 18] and computer vision [22, 31, 12] literature in the past with varying levels of success.

In this project, we intend to explore the performance of multi-task learning in conjunction with Curriculum [4] and On-Demand Learning [13]. This can be evaluated from two perspectives:

- Improving performance on individual tasks by jointly learning them with other tasks
- Developing generalizable representations for other target tasks (possibly with limited target data)

Multi-Task Curriculum Learning can be viewed from different perspectives. We could look at a curriculum among the tasks for Multi-Task Learning or look at a Curriculum within each task present in the array of tasks for Multi-Task Learning. We explore some of these perspectives in our experiments by looking at three different paradigms:

- 3D Representation Learning
- Image Restoration
- Image Classification

We next present each experiments for each problem domain in sections 2, 3 and 4. Finally, we provide some

insights into the applicability of Multi-Task Curriculum Learning and some directions we could explore in the future.

2. 3D Representation Learning

2.1. Prior work

Zamir *et al*. [35] explore learning general 3D representations from a set of fundamental 3D tasks such as 6 DoF Pose Estimation and Wide Baseline Feature Matching. The authors hypothesize that a generalizable 3D representation can be learned by jointly training on carefully selected foundational 3D tasks.

They first collect a large dataset called the Street View Dataset consisting of several target locations (defined by latitude, longitude and height). For each target location, multiple views of these locations collected by combining 3D models of the geographical locations and Google street view images (Figure 1).

Figure 1: Sample images from Street View Dataset [35]. Each row consists of multiple views of one target location (marked in red)

They propose a multi-task learning framework to jointly train a Siamese style network (Figure 2) on these tasks to learn a 3D image representation. They empirically show that the learned representation performs well on unseen regions for the source tasks (Pose Estimation, Matching). They also show that it generalizes well on other 3D tasks such as Scene Layout Estimation, Object Pose Estimation and Surface Normal Estimation. The learned representation is shown to cluster images based on the geometry (vanishing points) rather than the appearance.

2.2. Proposed Approach

We propose to extend the method from [35] to incorporate Curriculum Learning within the Multi-Task Learning framework. A curriculum is naturally defined for these tasks based on the baseline angle between the two views selected for pose estimation or matching. We define a 5 level curriculum as shown in Table 1. Some sample images from the different levels are shown in Figure 3. As we can see, the samples with smaller baselines are easier to match or identify the poses from. This could motivate us to naturally use the easy to hard curriculum strategy. But it is possible that the difficulty notions for humans are different from those of Deep Learning models. Based on these considerations, we consider different learning strategies in our experiments:

- **Rigid:** No curriculum incorporated. Each data batch consists of equal number of samples from each level.
- **Base:** This is a 2 level curriculum strategy used in [35]. The first half of training is performed on the data pairs with baseline angle less than 90 degrees. The second half proceeds with the full dataset.
- **Cumulative Curriculum (CC):** This is our proposed 5 level curriculum strategy as shown in Table 1. We initially sample only from Level-1. For the next set of epochs, we sample equally from Level-1 and Level-2. In a similar fashion, we keep adding the higher difficulty levels into the training process.
- **On-Demand Learning (ODL):** Adapting from [13], we sample data from the different levels based on the validation loss obtained at the end of K epochs. This allows the network to discover the difficulty of the various levels on it's own.

Note that the difficulty levels are defined commonly for both the tasks. At each iteration, we sample equal number of data points for both the tasks based on the defined curriculum.

Levels	Baseline angles (degrees)
1	$0 \leq \theta < 30$
2	$30 \leq \theta < 60$
3	$60 \leq \theta < 90$
4	$90 \leq \theta < 120$
5	$120 \leq \theta$

Table 1: Curriculum levels defined

2.3. Experimental Settings

In [35], the authors mention that the dataset contains over 100M matching pairs of data. Due to our time and resource constraints, we restrict our experiments to a smaller subset of the provided data. Our dataset statistics are shown in Table 2. We resize the images to 192×192 and crop a 101×101 patch around the target point in the image. We use the same architecture shown in Figure 2. All the multi-task learning models are trained to approximately 200k iterations with batch-size of 250. We use the SGD optimizer with a momentum of 0.95 and learning rate of 3e-3 which is reduced by 2 after every 100k iterations. The gradient is

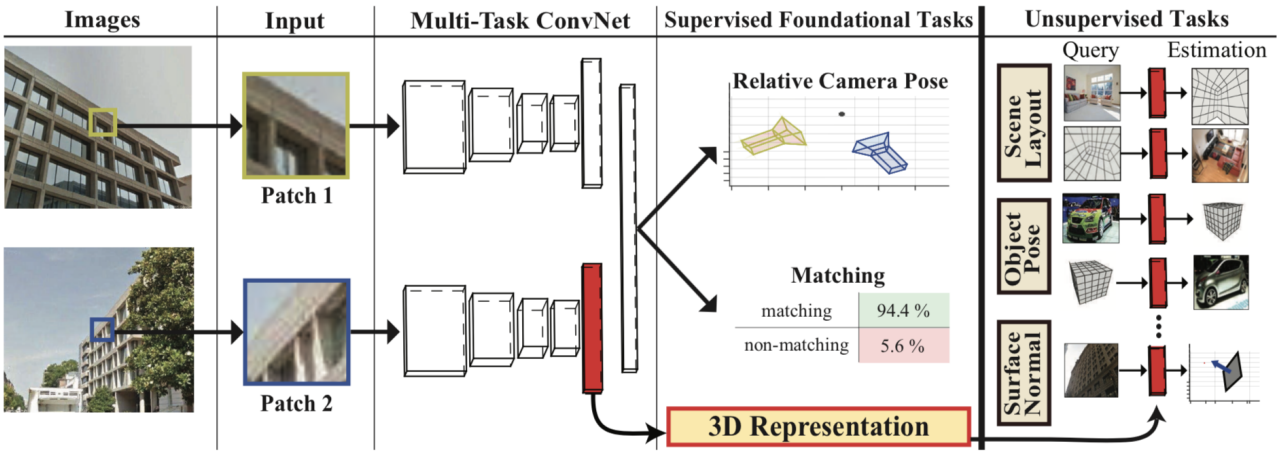


Figure 2: Siamese style network from [35] for jointly training on Wide Baseline Feature Matching and Pose Estimation

Figure 3: Each row shows a matching pair from the corresponding level

clipped at 0.1 based on the norm [1]. For pose estimation, the values we regress are the vector difference between the two poses. The difference is further normalized to have zero mean and unit standard deviation.

6 DoF Pose		Matching	
Split	# of samples	# of pos	# of neg
Train	872,979	872,979	1,760,551
Validation	5,000	5,000	5,000
Test	5,000	5,000	5,000

Table 2: Statistics of 3D representation learning dataset

2.4. Pose Estimation and Feature Matching

We jointly train the network using different learning strategies on Pose Estimation and Feature Matching. Feature Matching performance is measured using AUC (area under ROC curve) metric. The results are shown in Figure 4. As we can see, Joint ODL performs marginally better than all the other methods.

Pose Estimation is evaluated using two metrics. Average Angular Error measures the error in the pose angle predictions by computing norm of the relative angles between the relative pose prediction and ground truth, i.e.,

$$\text{Angular Error} = ||\hat{\theta}||_2^2$$

where $\hat{\theta}$ measures the relative rotation between the prediction and ground truth. Average Translation Error mea-

[1] http://pytorch.org/docs/master/_modules/torch/nn/utils/clip_grad.html

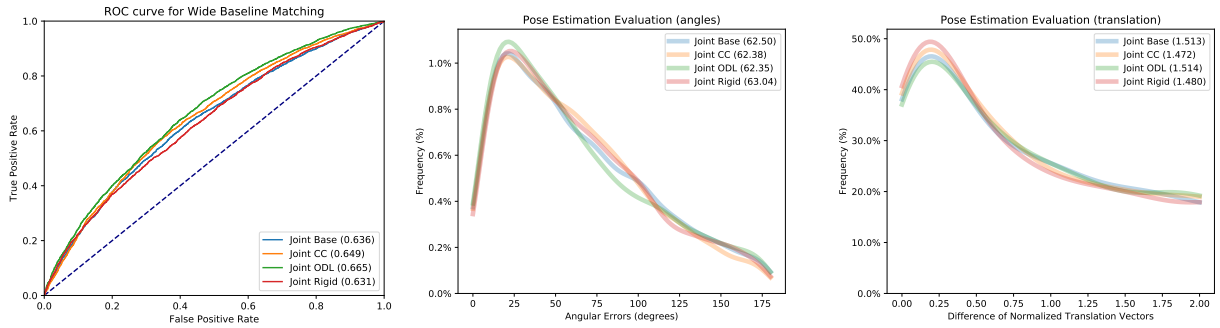


Figure 4: **(left)**: ROC plot on the Test set (AUC in legend) **(middle)**: Histogram statistics of Angular Errors (degrees) in Test (Average in legend) **(right)**: Histogram statistics of Normalized Translation Errors in Test (Average in legend)

sures the error in pose translation predictions by computing the L2 norm of the difference in normalized prediction and ground truth vectors. The results are shown in Figures 4. The Joint Rigid methods performs marginally better in Translation Error and Joint ODL performs marginally better in Angular Error. However, the differences in all the three results we have seen so far are marginal. We further explore the reasons for this in the next two sections.

2.5. Evaluating Generalization of Representation

As suggested in [35], we evaluate the learned representation on the auxiliary task of Surface Normal Estimation by keeping it fixed and learning a classifier on top. The NYU2 benchmark [27] is used and it consists of 795 training and 654 testing images. Surface Normals are predicted as a 20x20 grid. We cluster the Surface Normals from the training split as shown in Figure 5(a). A Delaunay Triangulation of the clusters is performed as shown in Figures 5(b), (c). The vertices of the triangles are the different clusters. Any surface normal can then be assigned to one of the triangles by performing triangulation. We perform a simple assignment strategy by assigning it to the closest vertex of the associated triangle. Surface Normal prediction is then converted to a classification task where the labels for prediction are the cluster labels. Evaluation metric is classification accuracy (shown as unbinned). Given that the dataset is biased towards certain surface normals, we also evaluate the classification accuracy per class and average them (shown as binned). Our results are shown in Figure 3. We also include the features from baseline networks trained on only Pose Estimation or Matching (Pose Base and Match Base). As we can see, most of the methods perform similarly and do not have very good classification accuracy. The results reported in [35] are 27.3 and 20.7 without and with binning respectively.

In order to understand the reasons for the comparable performances of the different methods and generally lower performance than reported, we perform an ablative study.

2.6. Ablative analysis of the Learned Representation

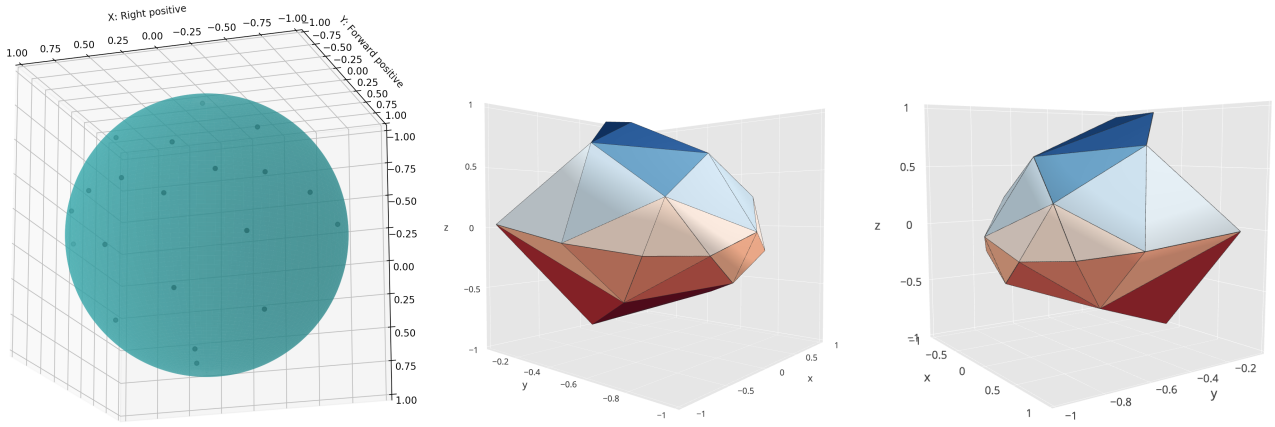
[35] visualize the image representations learned from different models using T-SNE [26]. Their proposed approach clusters the images based on the geometric structure present rather than appearance. We perform similar visualizations to evaluate the representations we learned (for Joint ODL) on the Street View dataset and on images from multiple Vanishing Point datasets [23, 6, 1]. As shown in Figures 6a, 6b, the clusters contain images which have similar vanishing points but varying appearance. However, we also noticed that the representations preferred to cluster images based on appearance (Figure 6c). This indicated that the models were not sufficiently trained.

2.7. Future Work

In the previous section, we empirically observed that the network was not sufficiently trained. We also noted that the validation performance had not saturated despite training for over 60 hours. Given that we trained on a single GPU (as opposed to 5-10 as suggested by [35]) for significantly lesser data samples (0.8M as opposed to 100M+ samples present in Street View) , we could not obtain better performance within our resource and time constraints. We also

	Without binning		With binning	
Model	Valid	Test	Valid	Test
Match Base	15.8	16.1	7.9	7.9
Pose Base	15.9	16.4	7.7	7.9
Joint Base	15.6	16.2	7.5	7.7
Joint ODL	16.3	**16.6**	8.2	7.8
Joint CC	16.8	**16.6**	8.4	7.9
Joint Rigid	15.6	16.4	7.7	**8.0**

Table 3: Surface Normal Estimation (accuracy) on NYU v2 dataset



(a) Surface Normal Clusters (b) Delaunay Triangulation of Clusters (c) Delaunay Triangulation of Clusters

Figure 5: Appearance based clusters from 3D Street View

noticed that our data was very noisy. It may be possible to reduce the noise and mine more quality data. Accordingly, our future work would be to:

- Implement multi-GPU training and scale up the system to train on more samples for more epochs
- Refine the dataset further by mining quality images with less noise
- Conclusively verify the efficacy of incorporating different curriculum strategies for this task

3. Image Restoration

We further examined our hypothesis on tasks from the Image Restoration. Specifically, we considered three tasks: Image Deblurring, Image Inpainting, and Pixel Interpolation. By training a single neural network jointly solving three tasks simultaneously, we expect that the performance on three tasks can increase.

3.1. Previous works

Deep learning has become popular methods for solving low-level vision tasks such as super-resolution and colorization [9], [5], [20]. One reason for its popularity is that collecting a dataset for those tasks is almost free, so it relaxes the biggest problem of deep learning: need for a large dataset. Researchers can corrupt the original images and use appropriate loss functions such as L_2 to train a deep neural network. This approach has shown some success in recent times for solving low-level vision problems.

However, Gao and Grauman [13] showed that the neural networks trained on these tasks fail when the levels of corruption changed from the training data. For instance, by reducing the corruption level for Image Inpainting (technically easier task), the networks trained on larger corruption levels were shown to fail. This demonstrates that a neural network is blindly solving problems and overfitting to the source dataset without learning sufficiently generalizable latent representations. This failure happens because a neural network is trained with fixed corruption levels. It is referred as *fixation* problem [13].

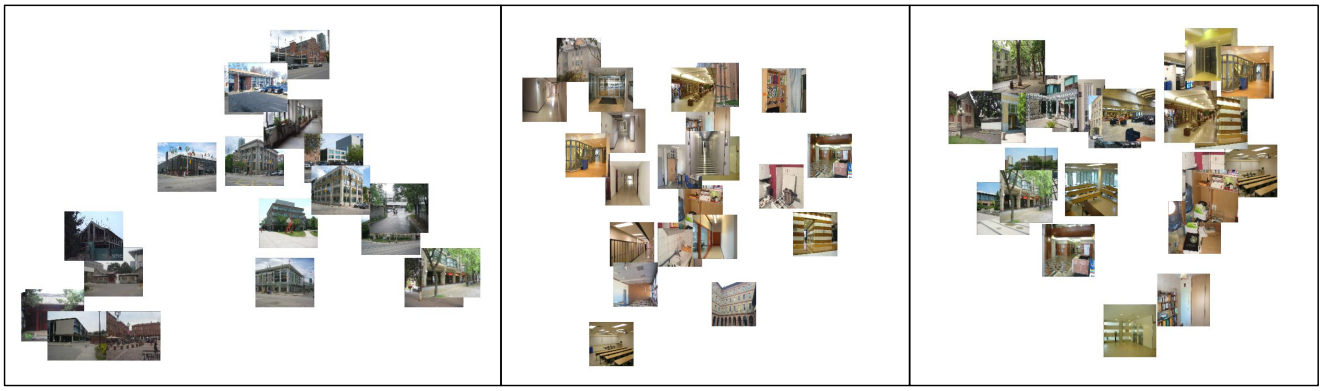
Gao and Grauman [13] showed that the fixation problem can be tackled by training the network on multiple difficulty levels with an On-Demand training schedule. We also attempt to solve the fixation problem, but our approach to use Multi-Task Learning by jointly training the networks across multiple tasks on a single difficulty level. We also use scheduled sampling between the different tasks based on On-Demand Learning.

3.2. Experiments

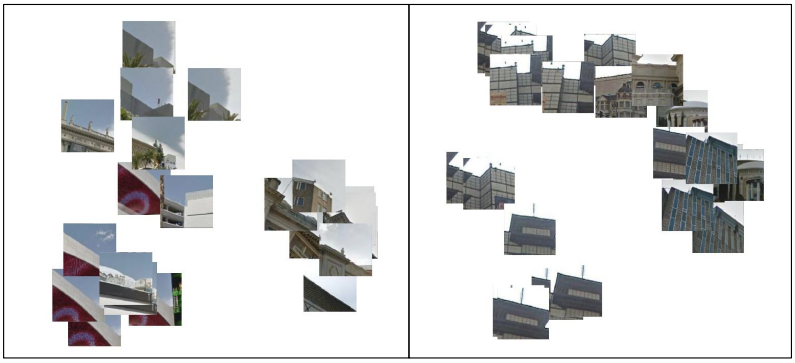
We mostly followed the experimental setting used in [13]. We used the same encoder-decoder architecture having 4 convolutional layers as an encoder and 4 transposed convolutional layers as a decoder. It is trained using the ADAM stochastic gradient descent algorithm with a batch size of 100.

The experiment is done with CelebFaces with Attributes (CelebA) dataset [24] for image deblurring, inpainting, and interpolation tasks. While the original work controls the intra-difficulty levels, we fixed the difficulty level in order to observe the effect of Multi-Task Learning only. Difficulty levels across all tasks are fixed at 1.

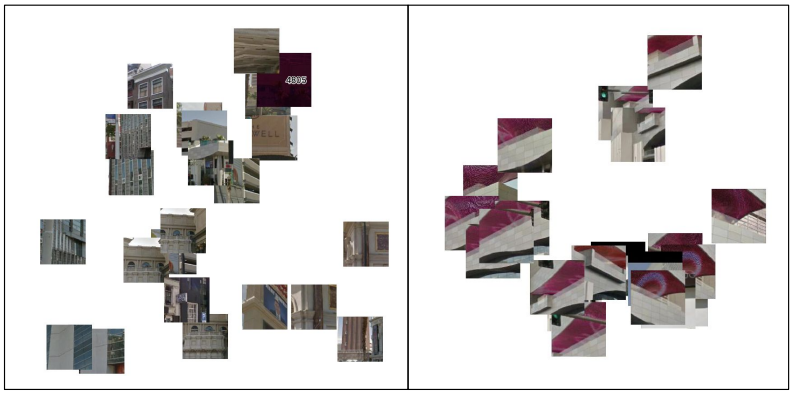
Two baselines are adopted. One is deep neural networks trained for each three task only. Another baseline is a Multi-Task trained network without any curriculum (**rigid**). These two will be compared with our proposed methods, Multi-Task On-Demand-Learning. (**multi-task ODL**). The quantitative results and qualitative results for three tasks are shown in Figure 7 and 8. The quantitative results are mea-



(a) Nearest Neighbours from Vanishing Point datasets



(b) Geometrically clustered samples from Street View dataset



(c) Appearance based clusters from Street View dataset

Figure 6: Cluster Analysis on two different datasets

sured with Peak Signal-to-Noise Ratio (PSNR), which is known to be a good measure of image quality.

Additionally, we also tested the combination of a jigsaw-puzzles task [29] and Inpainting. In this experiment, the encoder network is shared across tasks, while the parameters for decoder and puzzle solver are not shared. The results are drawn with a yellow line in Figure 7.

We can observe clear advantage of Multi-Task Learning on the Deblurring task. While the network trained only for Deblurring cannot handle clear images and emits noisy artifacts, the network trained with multiple tasks does not show such problems. The performance gaps between two methods are significant.

For the other two tasks, while the quantitative results indicate a slight decrease in the performance, we can see qualitatively better result for our method. In particular, the first two images on Inpainting demonstrate the benefit of Multi-Task Learning. When a minuscule box is given as the corruption, the single-task baseline just copied the given task image while our proposed method successfully recov-

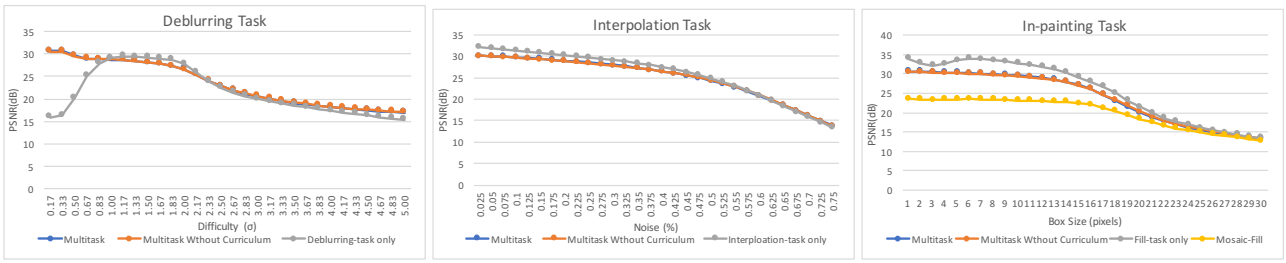


Figure 7: The quantitative results of three different tasks. For the deblurring task, σ ranging from 1.0 to 2.0 is used for training. The percentage of 0.15 to 0.30 is used for the interpolation task, and the box size of 7 to 12 is used for the inpainting task.

ers the original image. Similarly, when slightly larger image corruption is introduced (the second column of inpainting task), our results are visually better than the baseline. These results indicate that PSNR does not perfectly aligned with human perceptions.

However, we cannot observe any evidence supporting our hypothesis that On-Demand-Learning is beneficial. The resulting graph is almost identical whether a curriculum is used or not. We conjecture that it is due to the curriculum we choose, not the self-paced learning itself. We divide the number of samples in inverse proportional to PSNR score on validation set for a single batch, but the learning pace for all three tasks is similar to each other. Therefore, we could not observe a significant change in the curriculum; the number of samples for each task kept same during training. It was not the case for [13] since the learning pace varies across the different difficulty levels. It indicates that self-paced learning can be beneficial only when the learning pace varies across tasks.

4. Image Classification and Transfer

Finally, we examine the regularization effect of Multi-Task Learning. We are particularly interested in the regularization effect for Transfer Learning. We postulate that more general representation can be learned using Multi-Task Learning since a deep neural network is forced to solve multiple tasks with a limited set of parameters. It can induce the network to learn more generalizable features.

4.1. Experiments

In order to examine the hypothesis, we trained a CNN for digit classification tasks. Two commonly used dataset is used: MNIST [21] and Street View Housing Numbers (SVHN) [28]. We regard classifying digits for each dataset as two different tasks. Since the output of the network is same across the tasks, we can perform multi-task learning by training a network with two cross-entropy losses.

Similar to previous experiments, the performance of On-Demand-Learning is also tested. The number of samples for a batch is controlled by the losses for each task; a task having higher loss will be trained more with the larger number of training samples.

We examine generalization power on Omniglot dataset [19]. Omniglot dataset consists of more than 1600 characters from 50 different alphabets. It is usually used to test few-shot learning algorithms since it only contains 20 instance per character. Following the evaluation metric from few-shot learning literature [11], [32], we report the 5-way classification accuracy. Specifically, given single character instance and 5 example characters, we pick a character having minimum l_2 distance in the embedding space computed by pretrained networks. Therefore, we test transferability of features learned from MNIST or/and SVHN to Omniglot. The results are shown in Table 4.

Generally, regularization is enforced via data augmentation. It could be argued that a similar regularization effect can be achieved by simple data augmentation. In order to demonstrate that it is not the case, we also conducted a test with three different version of MNIST as suggested on [2].

4.2. Results

	Single Task		MNIST+SVHN	
	MNIST	SVHN	Rigid	ODL
MNIST	**0.989**	0.441	0.984	0.978
SVHN	0.215	**0.859**	0.850	0.844
Omniglot	0.459	0.452	**0.480**	0.477

Table 4: The results with MNIST and SVHN

We can see that Multi-Task Learning can generate more general features without hurting the performance on each base task. We were able to observe a boost of 2.1% in performance by training a neural network to jointly solve both tasks. The performance only slightly decreases on the original sub-tasks.

We can observe that data augmentation can provide regularizaing effect on MNIST, but it severely hurts the generalization power. The performance gain by augmenting data

(a) Deblurring Task

(b) Interpolation Task

(c) inpainting Task

Figure 8: Qualitative results on three different tasks with 10 difficulty levels (from easy to hard). From the top to row, given task images **(top)**, results of single-task baseline **(middle)**, results of multi-task baseline **(bottom)** is shown. The images are randomly chosen (not cherry-picked).

is only 0.001%, but the generalization performance drops significantly. The network still overfits on the dataset it is trained on. Therefore, we can conclude that datasets from different domains should be used to fully utilize the benefit of multi-tasks.

5. Conclusion

We explore using Multi-Task Curriculum Learning across various domains of Computer Vision such as 3D Representation Learning, Image Restoration, and Image Classification. In 3D Representation Learning, the effects of a Curriculum could not be clearly observed due to the limited training settings adopted. In contrast, we were able

	Single Task	MNIST +3 Variants	
	MNIST	Rigid	ODL
MNIST	0.989	0.989	**0.99**
MNIST AWGN	0.952	0.980	0.981
MNIST BLUR	0.970	0.986	0.987
MNIST RC AWGN	0.844	0.967	0.969
SVHN	**0.215**	0.180	0.146
Omniglot	**0.459**	0.445	0.419

Table 5: The results with MNIST dataset and its 3 variants [2]. AWGN and RC-AWGN stand for additive white Gaussian Noise and reduced contrast AWGN respectively. MNIST BLUR has the images filtered by simulated motion blurs.

to observe the performance gain through Multi-Task Learning in Image Restoration tasks. However, we could not observe any benefits of introducing a curriculum. Finally, in the experiments to confirm the effect on general representation learning, we verified that Multi-Task Learning can induce better feature representations. Though Multi-Task Learning itself can be beneficial, we feel that the straightforward method of introducing a curriculum between the tasks or within the tasks does not seem to be very effective. It is quite possible that each task requires a different approach for introducing curriculum as opposed to a universal strategy. We would like to explore some different ideas in this domain in the future.

References

[1] V. Angladon, S. Gasparini, and V. Charvillat. The toulouse vanishing points dataset. In *Proceedings of the 6th ACM Multimedia Systems Conference*, pages 231–236. ACM, 2015.

[2] S. Basu, M. Karki, S. Ganguly, R. DiBiano, S. Mukhopadhyay, S. Gayaka, R. Kannan, and R. Nemani. Learning sparse feature representations using probabilistic quadtrees and deep belief nets. *Neural Processing Letters*, 45(3):855–867, 2017.

[3] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. *Computer vision–ECCV 2006*, pages 404–417, 2006.

[4] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48. ACM, 2009.

[5] Z. Cheng, Q. Yang, and B. Sheng. Deep colorization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 415–423, 2015.

[6] P. Denis, J. H. Elder, and F. J. Estrada. Efficient edge-based methods for estimating manhattan frames in urban imagery. In *European conference on computer vision*, pages 197–210. Springer, 2008.

[7] C. Doersch, A. Gupta, and A. A. Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1422–1430, 2015.

[8] C. Doersch and A. Zisserman. Multi-task self-supervised visual learning. *arXiv preprint arXiv:1708.07860*, 2017.

[9] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2016.

[10] J. L. Elman. Learning and development in neural networks: The importance of starting small. *Cognition*, 48(1):71–99, 1993.

[11] C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. *arXiv preprint arXiv:1703.03400*, 2017.

[12] R. Gao and K. Grauman. On-demand learning for deep image restoration. In *ICCV*, 2017.

[13] R. Gao and K. Grauman. On-demand learning for deep image restoration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1086–1095, 2017.

[14] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, pages 10–5244. Manchester, UK, 1988.

[15] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[16] D. Jayaraman and K. Grauman. Slow and steady feature analysis: higher order temporal coherence in video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3852–3861, 2016.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[18] M. P. Kumar, B. Packer, and D. Koller. Self-paced learning for latent variable models. In *Advances in Neural Information Processing Systems*, pages 1189–1197, 2010.

[19] B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

[20] G. Larsson, M. Maire, and G. Shakhnarovich. Learning representations for automatic colorization. In *European Conference on Computer Vision*, pages 577–593. Springer, 2016.

[21] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[22] Y. J. Lee and K. Grauman. Learning the easy things first: Self-paced visual category discovery. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1721–1728. IEEE, 2011.

[23] B. Li, K. Peng, X. Ying, and H. Zha. Simultaneous vanishing point detection and camera calibration from single images. *Advances in Visual Computing*, pages 151–160, 2010.

[24] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 12 2015.

[25] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[26] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.

[27] P. K. Nathan Silberman, Derek Hoiem and R. Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012.

[28] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, page 5, 2011.

[29] M. Noroozi and P. Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*, pages 69–84. Springer, 2016.

[30] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, 2016.

[31] A. Pentina, V. Sharmanska, and C. H. Lampert. Curriculum learning of multiple tasks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5492–5500, 2015.

[32] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, and T. Lillicrap. One-shot learning with memory-augmented neural networks. *arXiv preprint arXiv:1605.06065*, 2016.

[33] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[34] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[35] A. R. Zamir, T. Wekel, P. Agrawal, C. Wei, J. Malik, and S. Savarese. Generic 3d representation via pose estimation and matching. In *European Conference on Computer Vision*, pages 535–553. Springer, 2016.

[36] R. Zhang, P. Isola, and A. A. Efros. Colorful image colorization. In *European Conference on Computer Vision*, pages 649–666. Springer, 2016.