ConvMTL: Multi-Task Learning via Self-Supervised Learning for Simultaneous Dense Predictions

Vijayasri Iyer[1][0000−0002−9143−7739], Senthil Kumar Thangavel[1][0000−0001−8160−7223], Madhusudana Rao Nalluri[2][0000−0002−5315−1932], and Maiga Chang[3][0000−0002−2827−6223]

[1] Department of Computer Science and Engineering, Amrita School of Computing, Coimbatore, Amrita Vishwa Vidyapeetham, India - 641112
thisisvij98@gmail.com
t_senthilkumar@cb.amrita.edu
[2] Department of Computer Science and Engineering, Amrita School of Computing, Amaravati, Amrita Vishwa Vidyapeetham, India - 522503
n_madhusudanarao@av.amrita.edu
[3] School of Computing Information and Systems, Athabasca University, Alberta, Canada
maigac@athabascau.ca

Abstract. Perception systems in autonomous vehicles are required to perform multiple scene-understanding tasks under tight constraints of latency and power. Single-task neural networks can become unscalable when the number of tasks increases in the perception stack. Multi-task learning has been shown to improve parameter efficiency and enable models to learn more generalizable task representations compared to single-task neural networks. This work explores a novel convolutional multi-task neural network architecture that simultaneously performs two dense prediction tasks, semantic segmentation and depth estimation. A self-supervised ResNet-50 backbone is used as the basis of the proposed network, along with a multi-scale feature fusion module and a dense decoder. The model uses a simple weighted loss function with an informed search algorithm identifying the optimal parameters. The performance of the proposed model on the segmentation task is assessed using the mean Intersection of Union (mIoU) and pixel accuracy. In contrast, absolute and relative errors assess the depth estimation task. The obtained results for segmentation and depth estimation are mIoU of 73.81%, pixel accuracy of 93.52%, an absolute error of 0.130, and a relative error of 29.05. The model's performance is comparable to existing multitask algorithms on the Cityscapes dataset, using only 2975 training samples.

Keywords: Multi-task Learning · Transfer Learning· Deep Learning · Computer Vision · Autonomous Driving

1 Introduction

The increased demand for intelligent transport systems has spurred the development of autonomous vehicles. Such a vehicle depends on its software system to make efficient, quick, and safe decisions on the road. The perception stack of an autonomous vehicle leverages recent advances in deep learning to make sense of the surroundings using sensors like Cameras, RADAR, and LiDAR. This safety-critical application must operate within the stringent cost and power limitations of the deployed hardware platform. The perception system performs hundreds of downstream tasks, which can make the single neural network per task setup unscalable. As a consequence, companies are slowly transitioning towards multi-task learning. Multi-task learning involves using a single neural network that simultaneously learns to solve multiple tasks. The Multi-task learning paradigm is considered a good fit for autonomous vehicle software systems for several reasons, including parameter efficiency, implicit data augmentation, and better generalisation of learned representations by a neural network [1]. Apart from the latency reduction at inference time, multiple works have also demonstrated that specific tasks learned by a neural network could benefit from sharing the model parameters and representations and learning from richer loss functions [2].

This work proposes a novel convolutional neural network architecture to perform two dense prediction tasks: semantic segmentation and depth estimation via transfer learning. Semantic segmentation and depth estimation allow for precise localization and navigation of an autonomous vehicle by enabling many downstream tasks, such as obstacle distance estimation, scene understanding, and reconstruction. In the domain of autonomous vehicles, the primary challenge lies not in the scarcity of data, but from the lack of labelled data, especially for multi-task learning. To address this limitation, we enhanced our multi-task learning framework by incorporating a backbone that is pre-trained using self-supervised learning [3]. Self-supervised learning is a methodology employed to train models in a supervised manner, using unannotated raw data, with the objective of learning intricate and informative representations. Lastly, in a multi-task learning setup, combining tasks and loss functions in a naive manner can be detrimental, leading to negative transfer, a form of destructive interference. This reduces accuracy across all tasks over the single neural network per task setup. We discuss how to deal with this issue using an informed search method for finding the optimal weight parameters that each loss should be assigned for smooth training. Our contributions are as follows :

1. A self-supervised approach to pre-training the encoder network (ResNet-50) of the multi-task model.
2. A novel decoder architecture that can perform semantic segmentation and depth estimation via transfer learning.
3. Using informed search optimization to determine the optimal weights for a weighted loss function.

The structure of our paper is as follows: Section 2 will cover previous works done in multi-task learning and self-supervised learning. Section 3 will cover the model architecture and the dataset in detail. Section 4 discusses our results and

the details for training and hyperparameter tuning. Section 5 is the conclusion, followed by references.

2 Related Work

2.1 Datasets

A multi-task learning dataset should ideally be annotated using the same input data for multiple tasks. The popular datasets that exist for multi-task learning on joint semantic segmentation and depth prediction are the Cityscapes dataset [4], the KITTI benchmark suite [5], Virtual KITTI [6], NYUD dataset [7] and the SUN RGBD Dataset [8]. Since Cityscapes [4] has balanced and synchronous annotations for real road scenes, we train and evaluate our proposed model using this dataset for both the semantic segmentation and depth estimation tasks. Due to the challenging nature of the dataset, most of the proposed multi-task models employ relatively complex task-weighting strategies and have a large number of parameters.

2.2 Multi-task Learning

Multi-task learning architectures can be categorized as hard-parameter sharing and soft-parameter methods. The hard-parameter sharing methods are generally more parameter efficient than their soft counterparts. Some of the popular soft parameter sharing methods include the MTAN network [9] with an attention-augmented SegNet, the GradNorm [10], PCGrad [11], CrossStitch [12] and Sluice Networks [13], AdaMT-Net [14]. The main challenge with these methods is that they either require meticulous tuning of hyperparameters, are hard to interpret or are computationally expensive. Simple approaches that primarily focus on the loss weighting, are the uncertainty weighting method [15], which dynamically balances the importance of different tasks based on via a homoskedastic loss function, MultiNet++ [16] that uses a geometric mean of the loss function to calculate weights. The XTasc-Net [17] and methods introduce attention structures between the decoders and MT-UNet [18] introduces an adaptive weight learning mechanism along with decoder cross-talk. Previous approaches following a similar network architecture to ours have been successful, including Quadronet [19] who propose a state-of-the-art, RetinaNet-based [20] architecture, delivering real-time results on four tasks of the Cityscapes dataset; however, the train their approach with a large annotated private dataset (1.2 million images), requiring significant labour and resources. Some noteworthy references such as the ParetoMTL [21] and NashMTL [22] consider the multi-task loss function as a multi-objective optimization problem. The uniqueness of our work is that we employ a simple loss weighting approach coupled with the use of Bayesian optimization search to find weight parameters, which we use for optimal weight tuning.

2.3 Self-supervised Multi-task Learning

The combination of self-supervised and multi-task learning has gained attention in recent years as it offers a potential solution to the challenge of acquiring labeled data for various tasks. The model can benefit from both the learned representations and the shared knowledge across tasks by pre-training a model on a self-supervised task and then fine-tuning it for multiple downstream tasks. The seminal approach on self-supervised learning is the SimCLR [23]. Other approaches such as the SimSiam [24], BYOL [25] and Masked Autoencoders [26] are used in conjunction with datasets of different sizes and different architectures. While traditional multi-task methods well explore the Cityscapes dataset, there has been limited exploration into efforts that leverage unlabelled data. The most popular work on multi-task self-supervised learning is [27] where the authors propose a ResNet101-based network. The work that is the most similar to ours is [28] where the authors propose a framework that uses self-supervised learning to improve performance on semantic segmentation and learning depth and colorization in a multi-task fashion using the ResNet18 model. Instead of using the Imagenet data for self-supervised pre-training, we propose a simple pre-training of the encoder using the extra training data from the same dataset for learning raw representations.

3 Methodology

The two most important components of a multi-task learning problem are the network architecture and the choice of the loss function. In this section, we discuss the preliminaries of the architecture and the architecture in detail. We also provide details of the loss function and the task weighting strategy used for our model.

3.1 Architecture

Our multi-task network architecture consists of the following components: an encoder, a multi-scale feature fusion module, two task-specific decoders and task heads.

Encoder We choose a ResNet-50 backbone having 23M parameters as our encoder network due to its superior performance on the ImageNet benchmark. The encoder model is trained in a self-supervised manner with extra training data provided in the Cityscapes dataset. The ResNet architecture was chosen for its training stability and ease of use in the self-supervised setting.

Multi-scale Feature Fusion The multi-scale feature fusion module consists of a BiFPN (Bidirectional Feature Pyramid Network) [29] that takes in features from various levels of the encoder as an input and aggregates them using

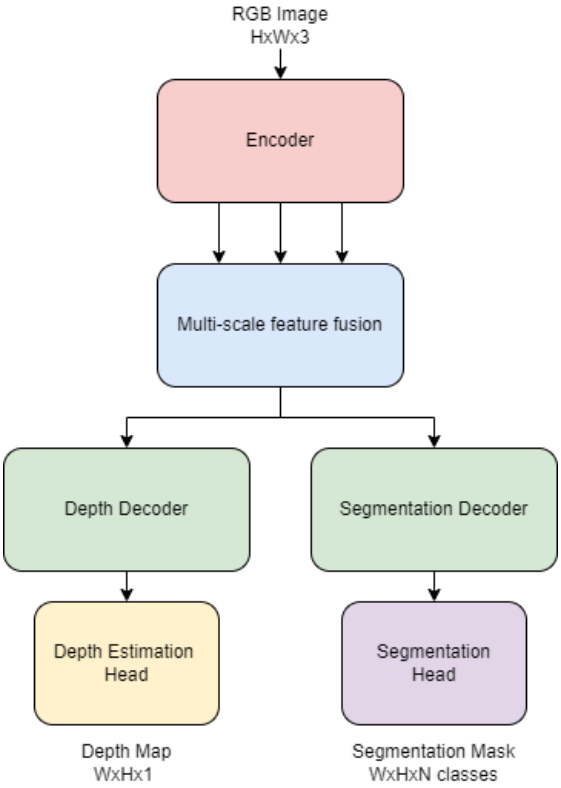


Fig. 1. ConvMTL Architecture

repeating pyramidal structures of neurons. The BiFPN model uses a series of bottom-up and top-down pathways to generate a rich set of features at multiple scales, which are then merged through the bidirectional connections to produce a more robust set of features that can be used for downstream perception tasks. The BiFPN also possesses a channel-wise attention mechanism, making it helpful in learning task-specific features.

Decoder The model has two task-specific decoders. The inputs from the BiFPN are sent to the decoder to learn task-specific features. The structure of the dense decoder is identical for both tasks. It consists of 5 upsampling blocks, each consisting of a convolution, a batch normalization layer and the Mish [29] activation function. The output from the decoder is then sent to the task-specific heads.

Task Heads The semantic segmentation head consists of a single convolutional layer followed by a bilinear upsampling layer and an output layer. The depth estimation head is identical in structure with the main difference being in the

number of output feature maps (N,C,H,W) where N is the number of classes, C is the number of channels and H,W is the height and width of the output map.

3.2 Self-supervised Pre-training

The encoder model is trained using the SimSiam [24] algorithm. In SimSiam, a Siamese neural network architecture is employed, consisting of two identical branches that share weights. The network is trained to predict the representations of augmented versions of the same input data while maximizing the similarity between the representations produced by the two branches. By maximizing the agreement between the augmented views of the same data, the algorithm encourages the model to learn meaningful and invariant features that capture important input data characteristics. Like SimCLR, SimSiam also uses a contrastive loss. However, the latter does not require negative samples and works well in situations with limited data. Figure 2 illustrates the SimSiam method.

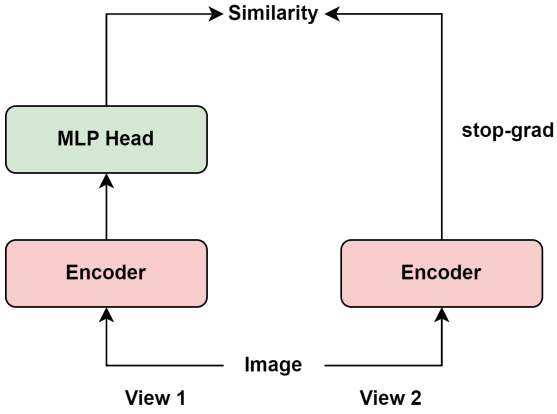


Fig. 2. SimSiam Architecture

3.3 Loss Function

Multi-task learning is considered to be a multi-objective optimization problem. Since a network is learning multiple tasks simultaneously, the resulting loss function required to be optimized has a complex search space compared to the individual task losses involved. As mentioned in the introduction, many methods have been proposed to find the optimal weights for a multi-objective weighted loss and optimal solutions for the loss search space. We experiment with a simple linear weighted loss function with α and β as the corresponding weight coefficients for each task loss. We use a simple pixel-wise cross-entropy loss shown in

Eq. function for the semantic segmentation task. The depth estimation task is performed with an L1 Norm loss shown in Eq..

$$L_{Seg} = -\sum_{c=1}^{M} y_{o,c} \log(p_{o,c}) \tag{1}$$

where c is the number of classes, o is the observation and

$$L_{Depth} = \sum_{i=1}^{M} \|y_{true} - y_{pred}\| \tag{2}$$

and

$$L_{Total} = \alpha \cdot L_{Seg} + \beta \cdot L_{Depth} \tag{3}$$

The deep learning model's convergence is highly dependent on the scales, convergence rates and complexities of the individual loss function. We use a Bayesian search optimisation method to find the weighted parameters α and β.

4 Experimental Results

This section discusses the dataset, evaluation metrics, implementation details, and the qualitative results for each task.

4.1 Dataset Description

The Cityscapes dataset [3] consists of high-resolution images of urban street scenes with corresponding annotations for a wide range of object and semantic classes collected across many European cities. The semantic segmentation annotations consist 2975 images, with 19 semantic class-level labels and 7 category-level labels. To avoid any issues of class imbalance, we choose the 7 category-level labels for our segmentation task. For depth estimation, a disparity map is provided, which is the inverse depth map, computed from stereo images using the SGM algorithm. For this task, we use the raw disparity map to avoid any artifacts created when converting the disparity to a depth map for the ground truth labels. The dataset also contains an extra training set consists of 19998 images with coarse annotations which we use as raw images for our self-supervised backbone.

4.2 Experimental Setup

The proposed network architecture was implemented in PyTorch. During training, we used the AdamW optimizer with an initial learning rate of 1e-3 and a weight decay of up to 1e-6. For the learning rate scheduler, we use cosine annealing with warm restarts to boost gradient updates in the network and prevent optimizing to a local minimum. The model was trained for 100 epochs with a

batch size of 4. The training was performed on an A100 GPU, taking approximately 5 minutes per epoch. The data were downsampled to an input size of 128×256 and augmented using the scales of 1.0, 1.2, 1.5 and random flips. All the parameters were initialized with a seed of 54321. The BiFPN, decoder and heads were initialized with 'Xavier Normal' weights.

SimSiam Training: We use a ResNet50 backbone with the initialized Imagenet weights for our contrastive pre-training. The network was trained using the 19998 extra training images provided in the dataset. The network was trained for 20 epochs with a batch size of 32 along with a scaled learning rate. An embedding dimension of 1000 is used along with a bottleneck dimension of 128.

Encoder: During the multi-task training phase, we use the pre-trained ResNet50 backbone. To provide more learning capacity to the model, we unfreeze the last block of the backbone network so that the network can learn a high-level generalized feature space.

BiFPN parameters: The BiFPN consists of 128 pyramid channels, 3 cell repeats, and convolutional channels of 512, 1024, and 204 for each repeat. The attention mode of the BiFPN was set to true, and the value of the epsilon parameter was set to 1e-5. The epsilon value of 1e-5 is suited for low-resolution images and objects of large sizes.

Decoder parameters: The dense decoder also had 128 pyramid channels and 128 segmentation channels with a dropout of 0.4 in the last layer of the decoder.

The total size of our network amounts to 30M parameters. The evaluation metrics used for the segmentation task are mean-intersection over union (mIoU) and pixel-wise accuracy. At the same time, the depth estimation task uses mean absolute error and relative error to measure the accuracy of the depth map. For the Bayesian search optimization to find alpha and beta, we use the Bayesian sweep method in the Weights and Biases SDK to perform our search with the total loss as the objective. The Lightly library was used to perform the contrastive pre-training.

4.3 Discussion

The ResNet50 model is trained in a self-supervised manner before it is trained in a multi-task learning setup. A scatterplot of the image embeddings encoded by the backbone model is shown in Figure 3. This indicates that the model has learnt good representations of the input data.

The proposed model is compared against the other multi-task learning methods mentioned in the introduction section that performs 7-class semantic segmentation and depth estimation on the Cityscapes dataset. The results are provided in Table 1. We also provide the results for the 19-class segmentation task with depth estimation in Table 2.

We also provide the qualitative results for both tasks in Fig 4. and Fig 5. With the Bayesian optimization method, we were able to examine the early impact of the choice of α and β on L_{total} as the main objective. We provide the parameter plots along with a parameter importance plot in Fig 1. and Fig 2. below. From the figure, we can see that the weight choice of α has a more significant impact on

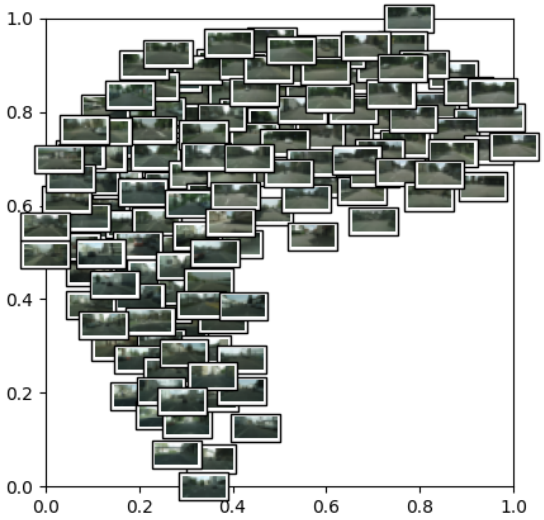


Fig. 3. Self-supervised Embeddings Scatterplot

performance. Moreover, lower values of α along with higher values of β lead to the better overall performance of the model. This can be attributed to the difference in the scales of the cross-entropy and L1 norm losses. Since the segmentation loss is of a higher scale, the value of the objective function tends to depend more on the segmentation task for convergence, leading to poor convergence on the depth estimation task.

We employ smaller batch sizes for the training and validation set. The training process of the network results in minor but constant oscillations during the process of loss convergence. Our model reported a training mIoU of 78.89% and 95.56% segmentation pixel accuracy along with 0.0078 absolute error and 20.33 relative error. The convergence rates of different tasks in multi-task learning can vary depending on various factors, such as the similarity of the tasks, the complexity of the network architecture, and the amount and quality of training data available for each task. In general, tasks that are more similar and share common features tend to converge faster than dissimilar tasks. Depth estimation is a more complex task that requires a longer convergence time but whereas segmentation which is a task that requires lower-level feature representations converges faster than tasks that require higher-level abstractions, such as scene-understanding or reasoning. Secondly, convergence also depends upon the bounds of the loss function. The segmentation loss and depth estimation loss have output values which are orders of magnitude apart from each other.

Table 1. Results: 7-class segmentation and depth estimation on the Cityscapes validation dataset.

Method Name	Segmentation		Depth Estimation	
	***mIoU*↑**	***Pix. Acc*↑**	***Abs. Err* ↓**	***Rel. Err* ↓**
STAN	51.90	90.87	0.0145	27.46
Dense	51.89	91.22	0.0134	25.36
CrossStitch	50.31	90.43	0.0152	31.26
MTAN	53.04	91.11	0.0144	33.63
PCGrad	53.59	91.45	0.0171	31.34
AdaMT-Net	62.53	94.16	0.0125	22.23
XTasc-Net	66.51	93.56	0.0122	19.40
MT-Unet	73.19	96.86	0.0137	27.50
Ours	**73.81**	**93.52**	**0.0130**	**29.05**

Table 2. Results: 19-class segmentation and depth estimation on the Cityscapes validation dataset.

Method Name	Segmentation		Depth Estimation	
	***mIoU*↑**	***Pix. Acc*↑**	***Abs. Err* ↓**	***Rel. Err* ↓**
MT-Unet	34.53	94.42	0.0142	28.75
Ours	51.31	90.56	0.0135	30.34

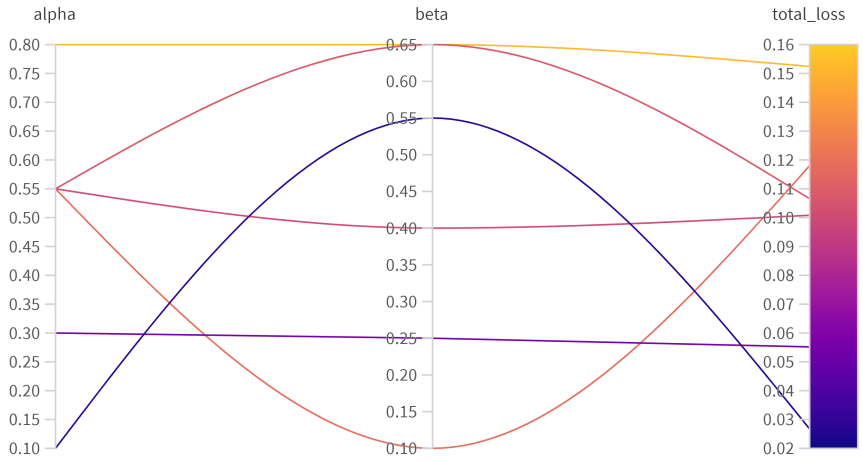


Fig. 4. Plot showing 6 early trials of Bayesian Search.

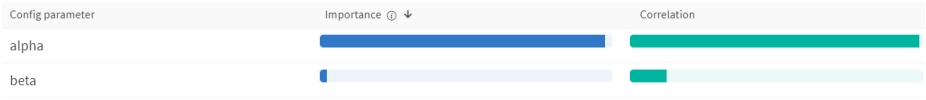


Fig. 5. Parameter Importance Plot.

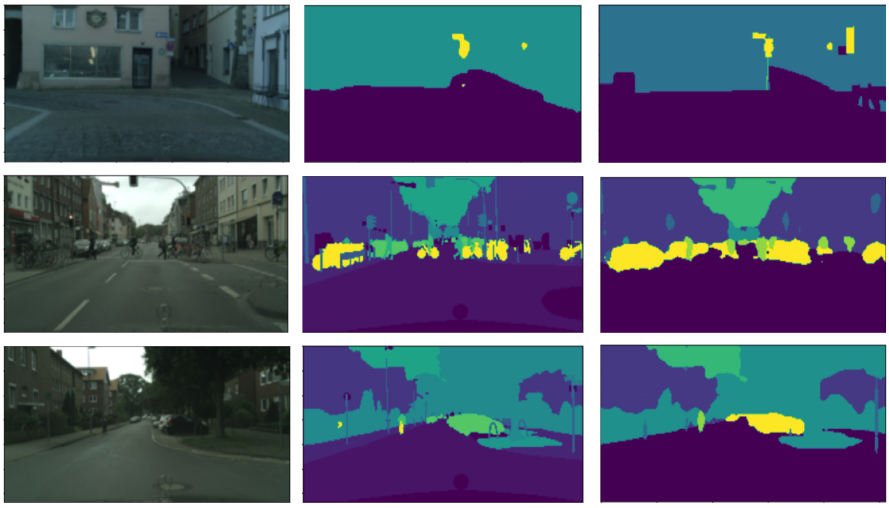


Fig. 6. 7-class Segmentation Results on the Cityscapes Validation Set.

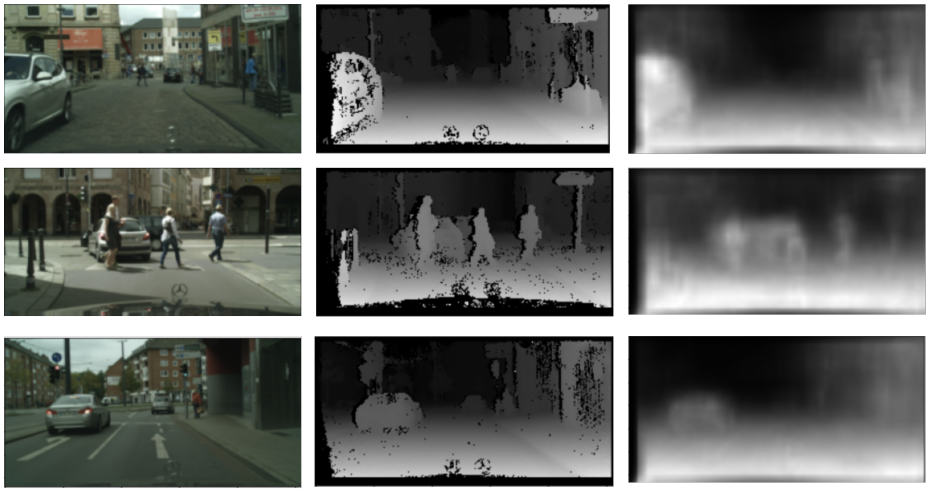


Fig. 7. Depth Estimation Results on the Cityscapes Validation Set

5 Conclusion

In this work, we have presented a self-supervised convolutional model capable of performing semantic segmentation and depth estimation simultaneously. We demonstrated how the encoder-decoder architecture is ideal for feature extraction and learning of task-specific features even with a very limited set of parameters. We also discuss the efficacy of a simple search based approach to find the optimal weights for a linear weighted loss function. Overall, our results indicate that the method achieves competitive performance with respect to the baseline multi-task algorithms on the Cityscapes dataset. Given that there has been limited research into the use of transfer learning and fine-tuning of models for joint segmentation and depth estimation, our approach offers a novel training framework that learns from labeled and unlabeled data and can be easily extended to other tasks, such as instance segmentation and optical flow estimation, by adding additional output heads to the network and extracting features at multiple levels of the multi-scale feature extraction module. The limitation of our model is that we have not tested it across multiple datasets, so there is limited evidence of domain adaptability. Further, exhaustive search trials need to be conducted to determine the exact impact of the coefficient choice and compared with optimization and other loss weighting methods. Further research can also be conducted to devise better supervised and unsupervised models, fine-tuning methods, larger datasets and task weighting schemes to achieve state-of-the-art and real-time performance.

References

1. Ruder, S.: An overview of multi-task learning in deep neural networks. arXiv preprint arXiv:1706.05098 (2017).
2. Standley, T., Zamir, A., Chen, D., Guibas, L., Malik, J., & Savarese, S. (2020). Which tasks should be learned together in multi-task learning?. In International Conference on Machine Learning (pp. 9120-9132). PMLR.
3. Balestriero, R., Ibrahim, M., Sobal, V., Morcos, A., Shekhar, S., Goldstein, T., Bordes, F., Bardes, A., Mialon, G., Tian, Y., Schwarzschild, A.: A Cookbook of Self-Supervised Learning. arXiv preprint arXiv:2304.12210 (2023).
4. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3213-3223 (2016).
5. Liao, Y., Xie, J., Geiger, A.: KITTI-360: A novel dataset and benchmarks for urban scene understanding in 2D and 3D. IEEE Trans. Pattern Anal. Mach. Intell. (2022).
6. Gaidon, A., Wang, Q., Cabon, Y. and Vig, E., 2016. Virtual worlds as proxy for multi-object tracking analysis. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4340-4349).
7. Couprie, C., Farabet, C., Najman, L., LeCun, Y.: Indoor semantic segmentation using depth information. arXiv preprint arXiv:1301.3572 (2013).
8. Song, S., Lichtenberg, S., Xiao, J.: SUN RGB-D: A RGB-D Scene Understanding Benchmark Suite. In: Proceedings of the 28th IEEE Conference on Computer Vision and Pattern Recognition (CVPR2015).

9. Liu, S., Johns, E., & Davison, A. J. (2019). End-to-end multi-task learning with attention. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 1871-1880).
10. Chen, Z., Badrinarayanan, V., Lee, C. Y., & Rabinovich, A. (2018). Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In International Conference on Machine Learning (pp. 794-803). PMLR.
11. Yu, T., Kumar, S., Gupta, A., Levine, S., Hausman, K., & Finn, C. (2020). Gradient surgery for multi-task learning. In Advances in Neural Information Processing Systems 33 (pp. 5824-5836).
12. Misra, I., Shrivastava, A., Gupta, A., & Hebert, M. (2016). Cross-stitch networks for multi-task learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 3994-4003).
13. Ruder, S., Bingel, J., Augenstein, I., & Søgaard, A. (2019). Latent multi-task architecture learning. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 33, No. 01, pp. 4822-4829).
14. A. Jha, A. Kumar, B. Banerjee and S. Chaudhuri, "AdaMT-Net: An Adaptive Weight Learning Based Multi-Task Learning Model For Scene Understanding," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 2020, pp. 3027-3035, doi: 10.1109/CVPRW50498.2020.00361.
15. Kendall, A., Gal, Y., & Cipolla, R. (2018). Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 7482-7491).
16. Chennupati, S., Sistu, G., Yogamani, S., & Rawashdeh, S. A. (2019). Multinet++: Multi-stream feature aggregation and geometric loss strategy for multi-task learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (pp. 0-0).
17. Nakano, A., Chen, S., & Demachi, K. (2021). Cross-Task Consistency Learning Framework for Multi-Task Learning. arXiv preprint arXiv:2111.14122.
18. A. Jha, A. Kumar, B. Banerjee and S. Chaudhuri, "AdaMT-Net: An Adaptive Weight Learning Based Multi-Task Learning Model For Scene Understanding," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 2020, pp. 3027-3035, doi: 10.1109/CVPRW50498.2020.00361.
19. Goel, K., Srinivasan, P., Tariq, S., and Philbin, J., "Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 315-324, 2021."
20. Lin, T.Y., Goyal, P., Girshick, R., He, K. and Dollár, P., 2017. Focal loss for dense object detection. In Proceedings of the IEEE international conference on computer vision (pp. 2980-2988).
21. Lin, X., Zhen, H.L., Li, Z., Zhang, Q.F. and Kwong, S., 2019. Pareto multi-task learning. Advances in neural information processing systems, 32.
22. Navon, A., Shamsian, A., Achituve, I., Maron, H., Kawaguchi, K., Chechik, G. and Fetaya, E., 2022. Multi-task learning as a bargaining game. arXiv preprint arXiv:2202.01017.
23. Chen, T., Kornblith, S., Norouzi, M. and Hinton, G., 2020, November. A simple framework for contrastive learning of visual representations. In International conference on machine learning (pp. 1597-1607). PMLR.
24. Chen, X. and He, K., 2021. Exploring simple siamese representation learning. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 15750-15758).

25. Grill, J.B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M. and Piot, B., 2020. Bootstrap your own latent-a new approach to self-supervised learning. Advances in neural information processing systems, 33, pp.21271-21284.
26. He, K., Chen, X., Xie, S., Li, Y., Dollár, P. and Girshick, R., 2022. Masked autoencoders are scalable vision learners. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 16000-16009).
27. Zhang, J., Zhang, H., Wang, Y., Sun, Q., Gao, J., Li, J. (2022). Boosting semantic segmentation with multi-task self-supervised learning for autonomous driving applications. arXiv preprint arXiv:2204.08412.
28. Doersch, C., Zisserman, A.: Multi-task self-supervised visual learning. In: IEEE International Conference on Computer Vision (ICCV), pp. 2051–2060 (2017).
29. Tan, M., Pang, R., and Le, Q. V., "Efficientdet: Scalable and efficient object detection," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10781-10790, 2020.
30. Misra, D., 2019. Mish: A self regularized non-monotonic activation function. arXiv preprint arXiv:1908.08681.