U-Net Transfer Learning for Image Segmentation with Various Encoders

Masashi Asai, Juexi Shao, Aaron Thompson, Kexuan Wang, and Lize Zhao

1 Introduction

Image segmentation is the process of dividing an image into separate groups of pixels based on their given labels. It is used in a wide array of applications from medical imaging to self driving cars. With the recent advancements in computing power, convolutional neural networks have made a significant impact on image segmentation, becoming widely used [1]. One particular CNN model that has proven to be highly successful is the U-Net model [2]. This U-shaped architecture has proven to be accurate in various different forms of image segmentation, such as satellite imagery [3] and medical imagery [4], while also winning numerous competitions, such as the Kaggle:Carvana Image Masking Challenge [5].

Recently, transfer learning has been used to improve upon the standard, single task, image segmentation. Transfer learning, a form of multi-task learning, is the process of transferring knowledge from a source setting to our target task or domain [6]. This usually involves first learning on one set of data, before applying the model to another set of data. By leveraging what was learnt from our source, we are able to overcome problems such as not having a large enough data sample. Further, through transfer learning, the model is often more generalisable to data that it wasn't trained on, making it more resilient.

In medical imaging, [7] found, in a review of transfer learning in medical imagery, that such a technique helped to reduce the training time and improve the segmentation accuracy when the target task is challenging and based on a smaller dataset. Meanwhile, [8] found that transfer learning with fine tuned CNNs outperformed the standard single task experiment.

One transfer learning technique for image segmentation that has shown promise recently is that involving the U-Net. Namely, that of replacing the encoder of the U-Net with that of a pre-trained network. [9], [5], [10] and [11] each replaced the U-Net encoder with a pre-trained VGGNet encoder to significant success in medical imagery. Meanwhile in remote sensing imagery, [12] combined a U-Net with a pre-trained DenseNet and was able to outperform the next best performing model. For the display of commodity in e-commerce, [13] combined a U-Net with a pre-trained ResNet and was able to outperform each of their comparison models in commodity semantic segmentation.

Thus, there are a number of instances of the effectiveness of U-Net transfer learning in image segmentation. In medical imaging in particular, there is a recent, growing prevalence of carrying out image segmentation using a U-Net model with the encoder replaced with a pre-trained VGGNet. In [9], they constructed their V-Unet in such a way, using the VGG16, pre-trained on the ImageNet dataset with the encoder parameters fixed after pre-training. By applying

their V-Unet model to the image segmentation of ribs from lung ultrasound images, they were able to achieve the high Dice score of 0.8632. In this paper, we are investigating if such a V-Unet model is applicable to more diverse datasets. To do this, we are applying the V-Unet model from [9] to the Oxford-IIIT Pets Dataset for image segmentation and investigating if it outperforms our baseline model of the standard U-Net model without any pre-training. What's more, given that the parameters of the V-Unet encoder are fixed after pre-training, this raises the natural question of how well the model will perform if the parameters of the encoder were not fixed. Thus, for an ablation study, we will also create a fine-tuned V-Unet model (FT V-Unet) in which the parameters of the encoder are not fixed after pre-training.

While it has become increasingly common to substitute the encoder for a different network in a U-Net, to the best of our knowledge, there has been no cross-comparison of the different types of possible encoders used in U-Net transfer learning for image segmentation. In fact, often times, the justification for the type of network used as an encoder is scant. As a further study, we therefore move beyond the V-Unet model and evaluate how the U-Net architecture performs with different pre-trained networks used as encoders. The aim is then twofold; to evaluate how well different networks perform as a U-Net encoder on the same dataset, and whether any of these improve upon the VGG16 encoder. These include the ResNet50 (R-Unet), DenseNet121 (D-Unet) and the MobileNetV2 (M-Unet). Each are pre-trained on ImageNet, with the parameters then fixed. A further point of interest is whether or not the performance of these networks as encoders correlates with their stand alone performance for image classification.

2 Methods

2.1 U-Net: Base Model

In our experiment, we are employing multiple different versions of the U-Net architecture, with different encoders used. Given this, to help in evaluating the performance of these different models, we are also using the standard U-Net as a baseline model. The architecture of this U-Net model follows closely to that given by [9], which itself follows that of the standard U-Net architecture [2]. That is, there is a contracting path and an expanding path. Each path involves blocks of two 3×3 convolutional layers, with each layer followed by a ReLU activation.

For the contracting path, there are five blocks, with a 2×2 max pooling with stride 2 between blocks. From the first to the fifth block, the number of channels double after each block, starting at 64 and finishing at 1024. For the expanding path, there are four blocks, each preceded by a 2×2 upsampling and a concatenation. The upsampling is done by using transposed convolutional layers, and halves the number of channels. The concatenation skip connections connect blocks of the same filter size between the contracting and expanding path and

allows the network to reuse features from the contracting path. Finally, a 1×1 convolutional layer with a sigmoid activation was used as the final layer.

2.2 V-Unet: U-Net architecture with VGG16 Encoder

For the V-Unet model, we are again following the same architecture as that used in [9]. That is, we use the standard architecture of the U-Net, only with the encoder replaced with the VGG16. The VGG16 architecture used follows closely to that of the standard VGG16 as given by [14]. That is, a series of blocks containing multiple 3×3 convolution layers, with each layer followed by a ReLU activation. There are two blocks of two layers and three blocks of three layers, with a 2×2 max pooling layer in between blocks. The key change is that the final three dense layers normally found in the VGG16 are removed. The expanding path is the same as that for the standard U-Net as previously described. (See Appendix for a graph of the V-Unet architecture as given by [9]).

2.3 Additional Networks

As well as using the VGG16 as the encoder, we also look to move beyond the architecture proposed by [9] and investigate the effects of using other established convolutional networks as the encoder in a U-Net architecture. For each network encoder, we integrate them into a U-Net architecture in the same manner as was done for the V-Unet. That is, we use their standard network architecture as the encoder, only with the final output layers removed. For this experiment, we use the following,

- **D-Unet**: A U-Net architecture with a DenseNet121 encoder. The DenseNet121 architecture as described by [15] was used, minus the final pooling layer.
- **R-Unet**: A U-Net architecture with a ResNet50 encoder. The ResNet50 architecture as described by [16] was used, minus the final pooling layer.
- **M-Unet**: A U-Net architecture with a MobileNetV2 encoder. The MobileNetV2 architecture as described by [17] was used, minus the final pooling layer.

3 Experiments

3.1 Experimental Design

For the first experiment, we evaluate the performance of the transfer learning of the V-Unet model. To do this, we compare the the V-Unet model against the U-Net model. The parameters of the V-Unet encoder are pre-trained and fixed after pre-training. The U-Net model meanwhile has had no pre-training and acts as our baseline model. To look closer at the effects of transfer learning, we will also build a version of the V-Unet with the parameters of the encoder not fixed, and so tuned further on our data, (FT V-Unet). We then compare this to our V-Unet with fixed encoder parameters and the baseline model.

For the second experiment, we carry out a cross comparison using different pre-trained networks as the U-Net encoder. These include the R-Unet, D-Unet and M-Unet as described in section 2. The parameters of each encoder will be fixed after pre-training. We then compare against the V-Unet.

To pre-train the encoders of our models, we use ImageNet as the training data. Since the encoders we use, such as VGG16, ResNet50, etc., are models that have been used in various researches, pre-trained models are available online or through machine learning frameworks. In this analysis, we constructed the models by importing the respective trained models from Keras.

In order to train and evaluate the performance of these models, we train each model for 50 epochs, with a learning rate of 1e-5 and using an Adam optimiser. We check the performance of the model on validation data at each epoch.

For the performance metrics, we employ the Dice coefficient (DICE)[18], as the main metrics for the evaluation. This metric is the ratio between the average number of elements in the two sets and the number of common elements of those two (in image segmentation, the ratio of common elements between the prediction and the ground truth). It is a commonly used indicator for image segmentation tasks. Furthermore, to check the robustness of the results, we also calculate the Jaccard index[19], or IOU. This is the ratio between the number of elements in the union of the two sets and the number of common elements. Finally, we also calculate the per-pixel binary accuracy. For the loss function, since the DICE takes a real value, we use the negative value of it.

3.2 Dataset

We use a simplified dataset created by preprocessing the Oxford-IIIT Pet Dataset, which is an open source database for image classification, segmentation and object detection containing 256×256 pixel RGB cats and dogs images. The dataset used in this analysis consists of training and validation data and each includes 2210, 738 samples respectively. In the segmentation task, only the animals (cats and dogs) and the background in the image are distinguished, and whether the animal is a cat or a dog is not the target of the distinction. (See Appendix B (6) for a summary of the dataset).

4 Results

Table (1) summarises the results of the two experiments above listing the best result (with highest DICE) in 50 epochs. We also graph the the performance of each metric during the training process in the appendix.

As can be seen from these results, in the comparison of Unet and V-Unets, the DICE for the U-net (3), V-Unet (4) and FT V-Unet (5), were 0.8327, 0.8989 and 0.9345 respectively. Given both V-Unets experienced such an improvement over the baseline model, this indicates that transfer learning had a positive impact on image segmentation for our animal dataset. The same is true for IOUs and binary accuracy. What's more, by fine tuning the encoder parameters, the FT

Model	Encoder	DICE	IOU	Accuracy
Unet (base)	None	0.8327	0.7141	0.8605
V-Unet	VGG-16	0.8989	0.8167	0.9160
V-Unet(ablated)	VGG-16(Unfixed)	0.9345	0.8773	0.9461
D-Unet	DenseNet121	0.8400	0.7246	0.8730
R-Unet	ResNet50	0.7569	0.6097	0.7858
M-Unet	MobileNetV2	0.8014	0.6692	0.9053

Table 1. Results with various encoders

V-Unet model was able to significantly outperform the V-Unet. This indicates that further fine tuning with transfer learning can lead to a further improvement in performance. The learning process shows that the speed of learning is significantly higher for the transfer learning models, with a trend towards convergence in performance even after less than 10 epochs. This can be interpreted as a result of the effectiveness of the features values obtained from the pre-trained auxiliary task in the transfer learning. On the other hand, we can see that over-fitting started at a much earlier stage in FT V-Unet.

Meanwhile, the results of the comparison between the models using different encoders are somewhat difficult to interpret: as far as the DICE is concerned, the performance is V-Unet, D-Unet (6), M-Unet (8) and R-Unet (7) in that order, and the variation in scores is also relatively high. From these results, it seems that for models other than V-Unet, replacing the encoder does not necessarily have a positive effect. Also, the learning process tended to be different for each model, with M-Unet still showing an improvement in performance at 50 epochs, while for R-Unet there is a trend towards convergence (while with low performance). For D-Unet, the performance improvement is stepwise, which can be seen as almost being trapped in local minima or saddle points.

In addition, for the purpose of examining the performance of the models in more detail, we created the Figure 1 below showing a visualization of the predictions of each model. In the figure, in order to emphasise the comparison of the predicted values between the models, the real values between 0 and 1 predicted for each pixel are used when we create each image (i.e. the numbers are not assigned to a binary value of 0 to 1). The results show that the visual prediction accuracy generally follows what the models' metrics imply, with the FT V-Unet in particular performing fairly well. On the other hand, for D-Unet and M-Unet, the model predicts a large part of the background to be around 0.5 or even near 1, indicating they have a particular and undesirable way of learning.

5 Discussion

As shown in Table 1, it is notable that, despite the relatively small number of training epochs, the results of the comparison for U-net, V-Unet and FT V-Unet clearly shows the positive impact of transfer learning and fine-tuning in all of the metrics. From the graphs in the appendix, the FT V-Unet and V-Unet both converged very quickly (Tables 5 and 4) after only a relatively few epochs. Meanwhile the validation loss of the U-Net shows signs of still improving, suggesting further epochs may be required (Table 3).

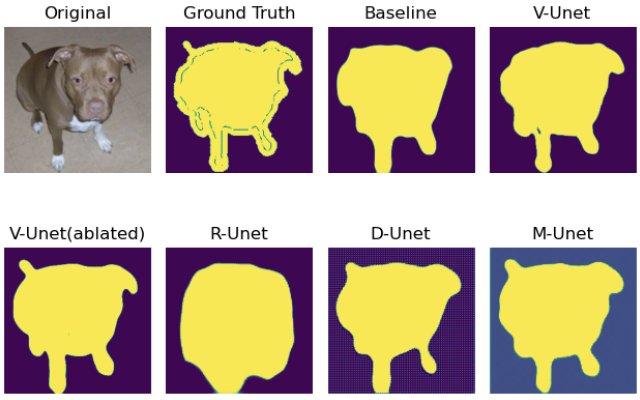

Fig. 1. Visualized Predictions

The comparison between the different encoders is more interesting. While the ResNet and DenseNet are more advanced and are supposed to outperform VGG in image classification tasks in general, R-Unet and D-Unet did not perform well in comparison with V-Unet. R-Unet was even inferior to M-Unet with MobileNet type encoder, which is characterized by small size, low latency and low power. Of course, these results were obtained in a limited experiment, so further verification of the results will be necessary in the future, not only by increasing the number of epochs, but also by experimenting with different optimisers and finer hyper-parameter tuning to avoid being trapped in local optima.

Finally, the results of the predicted images show that the training pattern in D-Unet and M-Unet is not favourable. This may indicate that there may have been a bias in predicting the whole image as an animal, as animals make up a large proportion of the total image in the dataset used here. Therefore, future analyses can be developed, in addition to the above points, such as the use of augmented data, the use of other loss functions, or the selection of more segmentation-specific auxiliary tasks.

6 Conclusion

In this study, we have applied the V-Unet model proposed by [9] to the Oxford-IIIT Pet Dataset, carrying out image segmentation. With this transfer learning technique, we were able to achieve a significant outperformance compared to the baseline U-Net model for each of the evaluation metrics used. However, by fine-tuning the encoder parameters further, we are able to see a significant improvement over the V-Unet model. This suggests that further fine tuning the encoder parameters after pre-training can lead to a significant improvement. In our cross-comparison of the U-Net transfer learning with different encoders, the VGG16 saw the best performance, while the ResNet50 saw the worst. Interestingly, the performance of the networks as pre-trained U-Net encoders did not correlate with their respective performance in image classification tasks. Although further verification of the results may be required for the cross-comparison.

Appendix

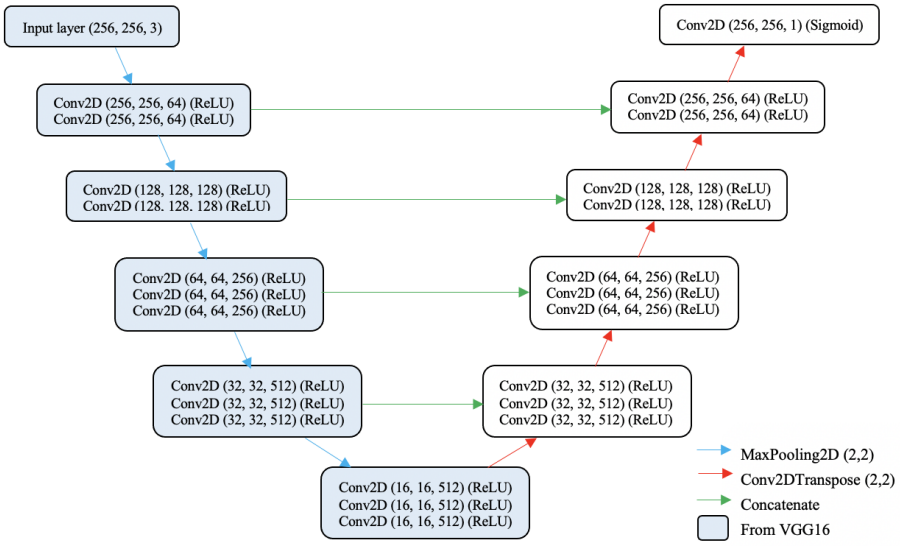

Fig. 2. V-Unet Architecture as given by [9]

	Number of samples	(cats samples)	(dogs samples)
Training data	2210	1475	735
Validation data	738	527	217
Test data	738	503	235

Table 2. Summary of Dataset

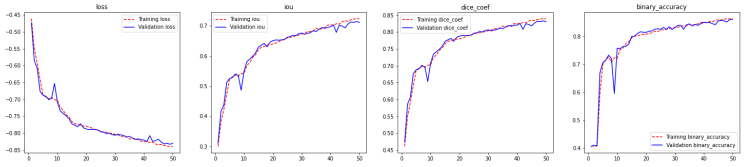

Fig. 3. Training Results for the Unet

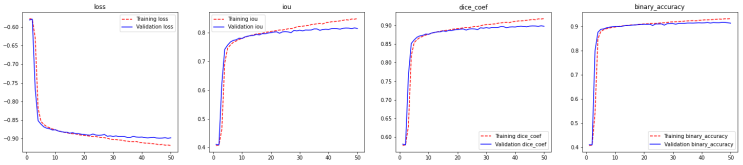

Fig. 4. Training Results for the V-Unet

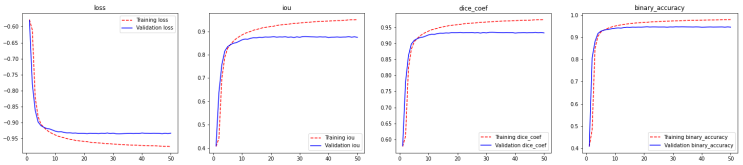

Fig. 5. Training Results for the ablated version of V-Unet

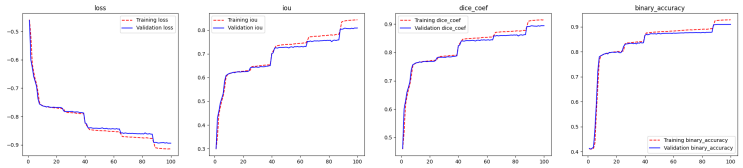

Fig. 6. Training Results for the D-Unet

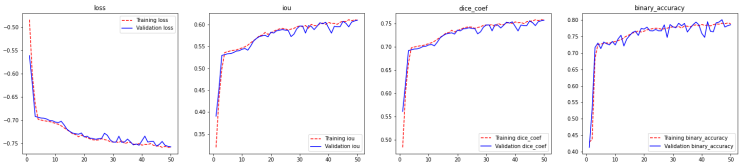

Fig. 7. Training Results for the R-Unet

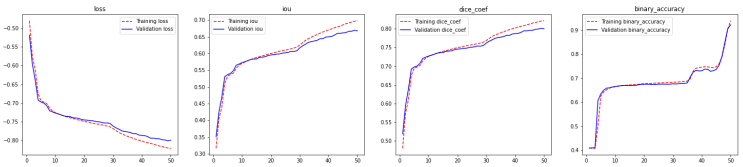

Fig. 8. Training Results for the M-Unet

References

1. Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. *CoRR*, abs/2001.05566, 2020.
2. Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015.
3. Vladimir Iglovikov, Sergey Mushinskiy, and Vladimir Osin. Satellite imagery feature detection using deep convolutional neural network: A kaggle competition. *CoRR*, abs/1706.06169, 2017.
4. Vladimir Iglovikov, Alexander Rakhlin, Alexandr A. Kalinin, and Alexey Shvets. Pediatric bone age assessment using deep convolutional neural networks. *CoRR*, abs/1712.05053, 2017.
5. Vladimir Iglovikov and Alexey Shvets. Ternausnet: U-net with VGG11 encoder pre-trained on imagenet for image segmentation. *CoRR*, abs/1801.05746, 2018.
6. Sebastian Ruder. Transfer Learning - Machine Learning's Next Frontier. http://ruder.io/transfer-learning/, 2017.
7. Davood Karimi, Simon K. Warfield, and Ali Gholipour. Critical assessment of transfer learning for medical image segmentation with fully convolutional neural networks. *CoRR*, abs/2006.00356, 2020.
8. Nima Tajbakhsh, Jae Y. Shin, Suryakanth R. Gurudu, R. Todd Hurst, Christopher B. Kendall, Michael B. Gotway, and Jianming Liang. Convolutional neural networks for medical image analysis: Full training or fine tuning? *CoRR*, abs/1706.00712, 2017.
9. Dorothy Cheng and Edmund Y. Lam. Transfer learning u-net deep learning for lung ultrasound segmentation, 2021.
10. Anindya Pravitasari, Nur Iriawan, Mawanda Almuhayar, Taufik Azmi, Irhamah Irhamah, Kartika Fithriasari, and Widiana Ferriastuti. Unet-vgg16 with transfer learning for mri-based brain tumor segmentation. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 18:1310, 06 2020.
11. Chirag Balakrishna, Sarshar Dadashzadeh, and Sara Soltaninejad. Automatic detection of lumen and media in the IVUS images using u-net with VGG16 encoder. *CoRR*, abs/1806.07554, 2018.
12. Binge Cui, Xin Chen, and Yan Lu. Semantic segmentation of remote sensing images using transfer learning and deep convolutional neural network with dense connection. *IEEE Access*, PP:1–1, 06 2020.
13. Zhengrong Wu, Like Zhao, and Haixiao Zhang. Mr-unet commodity semantic segmentation based on transfer learning. *IEEE Access*, 9:159447–159456, 2021.
14. Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
15. Gao Huang, Zhuang Liu, and Kilian Q. Weinberger. Densely connected convolutional networks. *CoRR*, abs/1608.06993, 2016.
16. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.
17. Mark Sandler, Andrew G. Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. *CoRR*, abs/1801.04381, 2018.
18. Kelly H Zou, Simon K Warfield, Aditya Bharatha, Clare MC Tempany, Michael R Kaus, Steven J Haker, William M Wells III, Ferenc A Jolesz, and Ron Kikinis. Statistical validation of image segmentation quality based on a spatial overlap index1: scientific reports. *Academic radiology*, 11(2):178–189, 2004.

19. Paul Jaccard. The distribution of the flora in the alpine zone. 1. *New phytologist*, 11(2):37–50, 1912.