Learning Morphological Feature Perturbations for Semi-Supervised Segmentation

Author Name1

Editors: Under Review for MIDL 2022

Abstract

We propose MisMatch, a novel consistency-driven semi-supervised segmentation framework which produces predictions that are invariant to learnt feature perturbations. MisMatch consists of an encoder and a two-head decoders. One decoder learns positive attention to the foreground regions of interest (RoI) on unlabelled images thereby generating dilated features. The other decoder learns negative attention to the foreground on the same unlabelled images thereby generating eroded features. We then apply a consistency regularisation on the paired predictions. MisMatch outperforms state-of-the-art semi-supervised methods on a CT-based pulmonary vessel segmentation task and a MRI-based brain tumour segmentation task. In addition, we show that the effectiveness of MisMatch comes from better model calibration than its supervised learning counterpart. We release our code here: https://github.com/moucheng2017/Learning_Morphological_Perturbation_SSL

Keywords: Semi-Supervised, Learn Augmentation, Attention, Feature Perturbation, Vessel Segmentation

1. Introduction

Medical image segmentation using deep learning requires expertly-labelled big data. Labels are scarce because manual labelling of medical images by experts is prohibitively expensive in both time and money. Semi-supervised learning (SSL) aims to tackle label scarcity by leveraging the unlabelled data. The success of SSL is built on the cluster assumption, which states that data points belonging to the same cluster are more likely to be in the same class (Cahpelle et al., 2006). It is also common to combine the cluster assumption with the smoothness assumption which presumes that data points are more dense in the centre of a cluster. An ideal example of the cluster assumption is shown in the beneath Fig.1, where one cluster is of circles and the other cluster is of triangles, representing two different classes of data points respectively. When the cluster assumption holds, the optimal decision boundary should lie in low-density region between the two clusters to separate the two different classes of data points.

Consistency regularisation based methods have recently achieved the state-of-the-art performances across different SSL image classification tasks, by forcing the model to produce perturbation-invariant predictions (Tarvainen and Valpola, 2017; Sohn et al., 2020; Berthelot et al., 2020; Athiwaratkun et al., 2019). We explain the intuition of why consistency regularisation works in SSL classification using Fig.1. In the setting of image classification, each data point in Fig.1 is an image. Let’s also define a perturbation on a data point as randomly changing its position. A consistency loss (e.g. mean-squared error) is defined as the difference between the predictions on two different perturbations of one data point. As shown in Fig.1(a), two different perturbations (red arrows) are applied on
Figure 1: Cluster assumption says that the optimal decision boundary lies in low-density regions. In addition, consistency regularisation can force the model to put a decision boundary in low-density regions. Here we have two clusters of circle data points and triangle data points. Arrow: perturbation. Dotted line: decision boundary. (a) when perturbations are applied on the red data point which lies in the high-density region, the two red perturbed data points will still be classified as in the same circle class, therefore the consistency loss will be zero; (b) when perturbations are applied on the blue data point which is in the low-density region, the two blue perturbed data points will have two different predictions, the difference between the two predictions will then contribute to the consistency loss to train the model. The perturbations on the data points on the cluster edge can across the low-density regions, leading to wrong classification results, which formulate an effective consistency loss value to drive the decision boundary towards low-density regions.

the red data point, as the red data point lies in the high-density region at the centre of the cluster of circles, the two red perturbed data points still stay in the same cluster, both will be classified as circles. The consistency loss on two same predictions will be zero, which won’t update the weights of the model. On the other hand, as shown in Fig.1(b), when two different perturbations (blue arrows) are applied on the blue data point which lies in the low-density region, the two corresponding predictions on the two perturbed blue data points will be different, leading to a consistency loss value which will be back-propagated to update the model. The consistency loss is essentially a regularisation which pushes the decision boundary towards the low-density region. Given the cluster assumption, the networks can thereby correctly classify the unlabelled data points.

However, the cluster assumption is violated in segmentation at image level (French et al., 2020). In segmentation, each data point in Fig.1 is a pixel and the optimal decision boundary is the contour of the object. Most importantly, in segmentation, pixels are uniformly distributed in a dense fashion across the whole image, resulting in hardly low-density regions between adjacent objects in the image, which clearly breaks the cluster assumption. Empirically, Fig.1 in (French et al., 2020) visually shows that in segmentation, the low-density regions of pixels do not align well with optimal decision boundaries, making the separation of different objects very difficult.

Fortunately, (Ouali et al., 2020) reported that low-density regions can be observed at the feature level, more importantly, at the feature level, low-density regions align well with the optimal decision boundaries (e.g. contours of the objects). It is obvious that feature level perturbations play a pivotal role in semi-supervised segmentation with consistency regularisation. We recall that as shown in Fig.1, in order to use consistency regularisation successfully, the perturbations should be able to across the low-density regions (contours
Learning Morphological Feature Perturbations for Semi-Supervised Segmentation of objects). Therefore, suitable feature perturbations for consistency regularisation should focus on the periphery of an object. The easiest way to perturb the features to across the contours is, perturbing the peripheral features along the orientation which is perpendicular to the tangent line of the object contour. Equivalently, the feature perturbations should be able to constrain the object shape. Naturally, we are inspired by the classic morphological operations such as dilation and erosion which respectively adds or removes the boundary pixels of an object, while reserving the object’s shape. Nevertheless, the existing morphological operations are operating at the image level with fixed structuring elements which are not suitable for feature perturbations. Different from previously proposed feature perturbations (Ouali et al., 2020) for semi-supervised segmentation, which utilise hand-crafted feature perturbations. We use end-to-end approach to learn optimal feature perturbation from the data because previous predefined feature perturbations might not be optimal and not generalise well across different tasks. In this paper, we introduce MisMatch, a deep framework using attention mechanisms, which learns to apply morphological perturbations at the feature level for a semi-supervised segmentation framework with consistency regularisation.

2. Methods

The overarching concept behind MisMatch is to leverage different attention mechanisms to respectively dilate and erode the foreground features, which are combined in a consistency-driven framework for semi-supervised segmentation. As shown in Fig.2, MisMatch is a framework which can be integrated into any encoder-decoder based segmentation architecture.

Morphological Operations At Feature Level. Whereas classical morphological operations change the boundary of the foreground at the image level, our network topology is designed to affect the feature level. We combine two concepts. First, results in (Wei et al., 2018; Chen et al., 2017; Luo et al., 2016; Xu et al., 2020b) showing that Atrous convolution can enlarge foreground features by increasing false positives on the foreground boundary. Second, results in (Luo et al., 2016; Xu et al., 2020b) showing that skip-connections can shrink foreground features. In combination, we can achieve learning-based feature perturbations with both Atrous convolution and skip-connections for consistency-driven semi-supervised segmentation.

Architecture of MisMatch. We use U-net (Ronneberger et al., 2015) as backbone due to its popularity in medical imaging. Our MisMatch (Fig 2) has two components: an encoder (fe) and a two-head decoder (fd₁ and fd₂). The first decoder (fd₁) comprises a series of Positive Attention Shifting Blocks, which dilates the foreground. The second decoder (fd₂) contains a series of Negative Attention Shifting Blocks, which erodes the foreground.

2.1. Positive Attention Shifting Block in fd₁

The Positive Attention Shifting Block (PASB) (Pink block in Fig.2) dilates foreground features with positive attention to the foreground. Each PASB has two parallel branches, namely the main branch and the side branch. The main branch is used for processing visual information and it has the same architecture with a decoder block in a standard U-net, which comprises two consecutive convolutional layers with kernel size 3 followed by
Figure 2: MisMatch learns confidence-invariant predictions on the foreground: decoder $f_{d1}$ dilates foreground features and decoder $f_{d2}$ erodes foreground features. The final prediction is the average between outputs of $f_{d1}$ and $f_{d2}$. Any encoder-decoder segmentation network could be used.

ReLU and normalisation layers. The side branch is used to generate a dilating attention mask to guide the main branch to enlarge its foreground features. To do so, the side branch uses two consecutive Atrous convolutional layers with kernel size 3 and dilation rate at 5, each followed by ReLU and a normalisation layer. In order to learn the magnitude of feature change at each pixel, we apply a Sigmoid function at the end of the side branch. We use element-wise multiplication of the output of the side branch with the output of the main branch to perturb the features of the main branch. We then apply a skip-connection on the perturbed main branch output to yield the final output of the PASB.

2.2. Negative Attention Shifting Block in $f_{d2}$

The Negative Attention Shifting Block (Purple block in Fig.2) erodes foreground features using negative attention to the foreground. Following PASB, we design the NASB again as two parallel branches. The main branch is the same with the one in the PASB. The side branch is similar with the main branch but with a skip-connection on each convolutional layer. We also apply a Sigmoid function on the output of the side branch to learn the perturbation magnitude at each pixel. Then we multiply the learnt eroding attention mask
from the side branch with the output of the main branch. We also apply a skip-connection on the perturbed main branch output to yield the final output of the NASB.

2.3. Loss Functions

We use a streaming training setting to avoid over-fitting on limited labelled data so the model doesn’t repeatedly see the labelled data during each epoch. For labelled data we apply a standard Dice loss (Milletari et al., 2016) with the output of each decoder. For unlabelled data we apply a mean squared error loss between the outputs of the two decoders. This consistency regularisation is weighted by hyper-parameter $\alpha$ between (0.0005 to 0.004).

3. Experiments

CARVE 2014 The Classification of pulmonary arteries and veins (CARVE) dataset (Charbonnier et al., 2015) has 10 fully annotated non-contrast low-dose thoracic CT scans. Each case has between 399 and 498 images, acquired at various resolutions between (282 x 426) to (302 x 474). 10-fold cross-validation on the 10 labelled cases is performed. In each fold, we split cases as: 1 for labelled training data, 3 for unlabelled training data, 1 for validation and 5 for testing. We have more than 2000 slices for testing. We only used slices containing more than 100 foreground pixels. We prepared datasets with differing amounts of labelled slices: 5, 10, 30, 50, 100. It is worthy to mention the most 100 slices is equal to about 10% of the whole available labelled data. We cropped 176 \times 176 patches from four corners of each slice. Full label training uses 4 training cases. Normalisation was performed at case wise.

BRATS 2018 BRATS 2018 (Menze et al., 2015) has 210 high-grade glioma and 76 low-grade glioma MRI volumes, each case containing 155 slices. We focus on binary segmentation of whole tumours in high grade cases. We randomly selected 1 case for labelled training, 2 cases for validation and 40 cases for testing. We have 6200 slices for testing. We centre cropped slices at 176 \times 176. For labelled training data, we discarded empty slices and extracted the first 20 slices containing tumours with areas of more than 5 pixels. To see the impact of the amount of unlabelled training data, we used different numbers of slices at 3100 (20 cases), 4650 (30 cases) and 6200 (40 cases) respectively. Case-wise normalisation was performed and all modalities were concatenated.

Experimental Settings We performed five sets of experiments/analysis: 1) comparisons with baselines including supervised learning and state-of-the-art semi-supervised learning approaches (Sohn et al., 2020; Tarvainen and Valpola, 2017; Chen et al., 2019; Ouali et al., 2020) using either data or feature augmentation; 2) investigation of the impact of the amount of labelled data and unlabelled data on MisMatch performance; 3) ablation study of the decoder architectures; 4) ablation study on the hyper-parameter, on the CARVE dataset using 5 labelled slices; 5) calibration analysis of MisMatch, cross-validation on the CARVE with 50 labelled slices.

3.1. Baselines

The backbone is a 2D U-net (Ronneberger et al., 2015) with 24 channels in the first encoder. To ensure a fair comparison we use the same U-net as the backbone across all baselines.
The first baseline utilises supervised training on the backbone, is trained with labelled data, augmented with flipping and Gaussian noise and is denoted as “Sup1”. To investigate how unlabelled data improves performance, our second baseline “Sup2” utilises supervised training on MisMatch, with the same augmentation. Because MisMatch uses consistency regularisation, we focus on comparisons with five consistency regularisation SSL methods: 1) “mean-teacher” (MT) (Tarvainen and Valpola, 2017), with Gaussian noise, which has inspired most of the current state-of-the-art SSL methods; 2) the current state-of-the-art model called “FixMatch” (FM) (Sohn et al., 2020). To adapt FixMatch for a segmentation task, we use Gaussian noise as weak augmentation and “RandomAug” (Cubuk et al., 2020) without shearing for strong augmentation. We do not use shearing for augmentation because it impairs spatial correspondences of pixels of paired dense outputs; 3) a state-of-the-art model with multi-head decoder (Ouali et al., 2020) for segmentation (CCT), with random feature augmentation including Dropout (Srivastava et al., 2014), VAT (Miyato et al., 2017) and CutOut (DeVries and Taylor, 2017), et al. This baseline is also similar to models recently developed (French et al., 2020; Ke et al., 2020); 4) a further recent model in medical imaging (Chen et al., 2019) using image reconstruction as an extra regularisation (MTA), augmented with Gaussian noise; 5) a U-net with two standard decoders, where we respectively apply traditional erosion and dilation on the features directly in each decoder, augmented with Gaussian noise (Morph)”. Our MisMatch model has been trained without any augmentation. See Appendix.A for details of training and implementation.

### 4. Results and Discussion

#### Table 1: MisMatch (MM) vs Baselines on CARVE. Metric is Intersection over Union (IoU): mean (std) under 10-fold cross validation. P values from Mann-Whitney U-Test against MisMatch. Red: best model. Blue: 2nd best model.

<table>
<thead>
<tr>
<th>Slices</th>
<th>Supervised</th>
<th>Semi-Supervised</th>
<th>MTA</th>
<th>MT</th>
<th>FM</th>
<th>CCT</th>
<th>Morph</th>
<th>MM</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>48.32±1.97</td>
<td>50.75±2.00</td>
<td>54.91±1.82</td>
<td>56.56±2.38</td>
<td>49.30±1.81</td>
<td>52.54±1.74</td>
<td>52.93±2.19</td>
<td>60.25±3.77</td>
</tr>
<tr>
<td>10</td>
<td>53.38±2.83</td>
<td>55.55±4.42</td>
<td>57.78±3.66</td>
<td>57.99±2.57</td>
<td>51.53±3.72</td>
<td>55.25±2.52</td>
<td>57.08±2.96</td>
<td>60.04±3.64</td>
</tr>
<tr>
<td>20</td>
<td>52.09±1.41</td>
<td>53.98±4.42</td>
<td>60.78±4.63</td>
<td>60.46±3.74</td>
<td>55.16±5.93</td>
<td>60.81±4.09</td>
<td>60.19±4.97</td>
<td>63.59±4.46</td>
</tr>
<tr>
<td>50</td>
<td>60.69±2.51</td>
<td>64.79±3.46</td>
<td>68.11±3.39</td>
<td>67.21±3.05</td>
<td>62.91±6.99</td>
<td>65.06±3.42</td>
<td>64.88±3.25</td>
<td>69.39±3.74</td>
</tr>
<tr>
<td>100</td>
<td>68.74±1.84</td>
<td>73.1±1.51</td>
<td>72.48±1.64</td>
<td>71.48±1.57</td>
<td>72.58±1.84</td>
<td>72.07±1.75</td>
<td>72.11±1.88</td>
<td>74.83±1.52</td>
</tr>
<tr>
<td>Param. (M)</td>
<td>1.8</td>
<td>2.7</td>
<td>2.1</td>
<td>1.88</td>
<td>1.88</td>
<td>2.54</td>
<td>2.7</td>
<td></td>
</tr>
<tr>
<td>Infer.Time(s)</td>
<td>4.1e-3</td>
<td>1.8e-1</td>
<td>7.2e-3</td>
<td>4.3e-3</td>
<td>4.3e-3</td>
<td>1.5e-1</td>
<td>8e-3</td>
<td>1.8e-1</td>
</tr>
<tr>
<td>P values</td>
<td>9.13e-5</td>
<td>1.55e-2</td>
<td>4.5e-3</td>
<td>4.3e-4</td>
<td>1.05e-2</td>
<td>1.8e-3</td>
<td>2.2e-3</td>
<td></td>
</tr>
</tbody>
</table>

#### Table 2: MisMatch (MM) vs Baselines on BRATS. Metric is Intersection over Union (IoU). Each model was trained 3 times. Red: best model. Blue: 2nd best model.

<table>
<thead>
<tr>
<th>Slices</th>
<th>Supervised</th>
<th>Semi-Supervised</th>
<th>MTA</th>
<th>MT</th>
<th>FM</th>
<th>CCT</th>
<th>Morph</th>
<th>MM</th>
</tr>
</thead>
<tbody>
<tr>
<td>3100</td>
<td>53.74±10.19</td>
<td>55.76±11.03</td>
<td>50.53±8.76</td>
<td>55.29±10.21</td>
<td>57.92±12.35</td>
<td>56.61±11.7</td>
<td>53.88±9.99</td>
<td>58.94±11.41</td>
</tr>
<tr>
<td>4650</td>
<td>53.74±10.19</td>
<td>55.76±11.03</td>
<td>47.36±6.60</td>
<td>58.32±12.07</td>
<td>54.29±9.69</td>
<td>56.94±10.93</td>
<td>55.82±11.03</td>
<td>60.74±12.96</td>
</tr>
<tr>
<td>6200</td>
<td>53.74±10.19</td>
<td>55.76±11.03</td>
<td>50.11±6.00</td>
<td>56.92±12.20</td>
<td>56.78±11.39</td>
<td>57.37±11.74</td>
<td>54.5±9.75</td>
<td>58.81±12.18</td>
</tr>
</tbody>
</table>

#### Segmentation Performance:
1) MisMatch consistently and substantially outperforms supervised baselines, e.g. 24% improvement over Sup1 on 5 labelled slices, CARVE;
2) MisMatch consistently outperforms previous SSL methods (Sohn et al., 2020; Tarvainen and Valpola, 2017; Chen et al., 2019; Ouali et al., 2020) in Table 1, across different data sets, e.g. statistical difference when 6.25% labels (100 slices comparing to 1600 slices of full label) are used on CARVE (Table 1); 3) data diversity of training data highly affects the testing performance (Fig 3), e.g. in fold 3 on CARVE, MisMatch outperforms the full label training, in fold 7 and 8, MisMatch performs on par with full label training, whereas in the rest folds, MisMatch performs worse than the full label training; 4) more labelled training data consistently produces a higher mean IoU and lower standard deviation (Table 2).

**Ablation Studies** We performed ablation studies on the architecture of the decoders of MisMatch (Fig4) with cross-validation on 5 labelled slices of CARVE: 1) “MM-a”, a two-headed U-net with standard convolutional blocks in decoders, this model can be seen as no feature perturbation, however, they are essentially slightly different because of random initialisation, we denote the decoder of U-net as \( f_{d0} \); 2) “MM-b”, a standard decoder of U-net and a negative attention shifting decoder \( f_{d2} \), this one can be seen as between no perturbation and learnt erosion perturbation; 3) “MM-c”, a standard decoder of U-net and a positive attention shifting decoder \( f_{d1} \), this one can be seen as between no perturbation and learnt dilation perturbation; 4) “MM”, \( f_{d1} \) and \( f_{d2} \) (Ours). As shown in Fig4, our MisMatch ("MM") outperforms other combinations in 8 out of 10 experiments and it performs on par with the others in the rest 2 experiments. We also tested \( \alpha \) at 0, 0.0005, 0.001, 0.002, 0.004 with the same experimental setting. The optimal \( \alpha \) appears at 0.002 in Table 4.

**MisMatch Is Better Calibrated** We conjugate that MisMatch utilises unlabelled images to improve model calibration (Guo et al., 2017), leading to better segmentation performance. Model calibration reflects the trustworthiness of the network predictions, which are crucial in clinical applications. Following (Guo et al., 2017), we set \( B_m \) as the
Figure 4: (a) Ablation studies on decoder architectures. (b) Performance differences between other possible decoder combinations against our used design. (c) Ablation study on the hyper-parameter alpha which weights the consistency loss. All experiments were performed on 5 labelled slices with CARVE with cross-validation.

Figure 5: Reliability diagrams of one testing image. Our model appears to be more trustworthy than the baseline. Blue: Confidence. Red: Accuracy. (a): Sup1 (Supervised learning with U-net). (b): Outputs of positive attention decoders. (c): Outputs of negative attention decoders. (d): Average outputs of the two decoders. The smaller the gap between the accuracy and the confidence, the better the network is calibrated. See Discussion and AppendixE for more.

subset of all pixels whose prediction confidence is in interval $I_m$. We define accuracy as how many pixels are correctly classified in each confidence interval. The accuracy of $B_m$ is:

$$\text{acc}(B_m) = \frac{1}{|B_m|} \sum_{i \in B_m} 1(\hat{y}_i = y_i).$$

Where $\hat{y}_i$ is the predicted label and $y_i$ is the ground truth label at pixel $i$ in $B_m$. The average confidence within $B_m$ is defined with the use of $\hat{p}_i$ which is the raw probability output of the network at each pixel:

$$\text{conf}(B_m) = \frac{1}{|B_m|} \sum_{i \in B_m} \hat{p}_i.$$

The plot for comparing the accuracy and confidence for each interval is called Reliability map (Fig.5), the gap between the accuracy and the confidence is called expected calibration error, the smaller the gap, the better the network is calibrated. As shown in Fig.5, MisMatch produces better calibrated predictions. See Appendix D for analysis of expected calibration error.

5. Conclusion

We propose MisMatch, a consistency-driven SSL framework with attention-based feature augmentation for semi-supervised segmentation of medical images. MisMatch promises strong clinical utility by reducing the number of training labels required by more than 90%: when trained on just 10% of labels, MisMatch achieves a similar performance (IoU: 75%) to models that are trained with all available labels (IoU: 77%). Future work will extend MisMatch to multi-class 3D tasks as a lot of medical images are volumetric.
References


Wenhui Cui, Yanlin Liu, Yuxing Li, Menghao Guo, Yiming Li, Xiuli Li, Tianle Wang, Xiangzhu Zeng, and Chuyang Ye. Semi-supervised brain lesion segmentation with an adapted mean teacher model. *Information Processing in Medical Imaging (IPMI)*, 2019.


Wenlong Hang, Wei Feng, Shuang Liang, Lequan Yu, Qiong Wang, Kup-Sze Choi, and Jing Qin. Local and global structure-aware entropy regularized mean teacher model for 3d left atrium segmentation. *International conference on medical image computing and computer assisted intervention (MICCAI)*, 2020.


Appendix A. Implementation

We use Adam optimiser (Kingma and Ba, 2015). Hyper-parameters are: $\alpha = 0.002$, batch size 1 (GPU memory: 2G), learning rate 2e-5, 50 epochs. Each complete training on CARVE takes about 3.8 hours. The final output is the average of the outputs of the two decoders. In testing, we take an average of models saved over the last 10 epochs across experiments. Our code is implemented using Pytorch 1.0 (Paszke et al., 2019).

Appendix B. Related Work

SSL in classification A recent review (Oliver et al., 2018) summarised different common SSL (Iscen et al., 2019) (Miyato et al., 2017) (Tarvainen and Valpola, 2017) methods including entropy minimisation, label propagation methods, generative methods and consistency based methods. Entropy minimisation encourages models to produce less confident predictions on unlabelled data (Grandvalet and Bengio, 2004) (Lee, 2013). However, entropy minimisation might overfit to the clusters of classes and fail to detect the decision boundaries of low-density regions (see Appendix E in (Oliver et al., 2018)). Label propagation methods (Iscen et al., 2019) (Lee, 2013) aim to build a similarity graph between labelled data points and unlabelled data points in order to propagate labels through dense unlabelled data regions. Nevertheless, label propagation methods need to build and analyse their Laplacian matrices which will limit their scalability. Generative models have also been used to generate more data points in a joint optimisation of both classification of labelled data points and generative modelling (Kingma et al., 2014). However, the training of such a joint model can be complicated and unstable. On the other hand, consistency regularisation methods have achieved state-of-the-art performances across different benchmarks, additionally, consistency regularisation methods are simple and can easily be scaled up to large data sets. Of the consistency regularisation methods, Mean-Teacher (Tarvainen and Valpola, 2017) is the most representative example, containing two identical models which are fed with inputs augmented with different Gaussian noises. The first model learns to
match the target output of the second model, while the second model uses an exponen-
tially moving average of parameters of the first model. The state-of-the-art SSL methods
(Berthelot et al., 2020) (Sohn et al., 2020) combines two categories: entropy minimisation
and consistency regularisation.

SSL in segmentation In semi-supervised image segmentation, consistency regularisa-
tion is commonly used (Xu et al., 2020a) (Li et al., 2020) (Cui et al., 2019) (Hang et al.,
2020) (Fang and Li, 2020) (French et al., 2020) where different data augmentation techniques
are applied at the input level. Another related work (Li et al., 2018) forces the model to
learn rotation invariant predictions. Apart from augmentation at the input level, recently,
feature level augmentation has gained popularity for consistency based SSL segmentation
(Ouali et al., 2020; Ke et al., 2020). Apart from consistency regularisation methods in med-
icinal imaging, there also have been other attempts, including the use of generative models
for creating pseudo data points for training (Chaitanya et al., 2019) (Chen et al., 2020)
and different auxiliary tasks as regularisation (Kervadec et al., 2019) (Chen et al., 2019).
Since our method is a new consistency regularisation method, we focus on comparing with
state-of-the-art consistency regularisation methods.

Appendix C. Visualisation of Features

Fig. 6 shows the visualisation of the features inside of the last positive attention shifting block
using CAM (Zhou et al., 2016)

Fig. 7 shows the visualisation of the features inside of the last negative attention shifting
block using CAM (Zhou et al., 2016).

Appendix D. Plots of Accuracy vs Calibration Error

See Fig. 8. It is obvious to see that with the help of unlabelled images, MisMatch achieves
better segmentation accuracy with lower calibration error.

Appendix E. More Reliability Maps

See Fig. 9.

Appendix F. Visual Results

See Fig. 10.
Figure 6: Features in the last positive attention shifting block. (a) Testing input image. (b) Ground truth. (c) Final output of the last positive attention shifting block. (d) Output of the main branch. (e) Output of the attention branch, attention mask. (f) Output of the main branch after multiplication with the attention mask from the attention branch. (g) Differences on the output before and after the attention mask is applied. It is obvious to see that foreground features are dilated in (f). As shown in (g), the learnt positive attention mask mostly increases the confidence around the periphery of foreground, as the confidence in the centre of the objects are already high.
Figure 7: Features in the last negative attention shifting block. (a) Testing input image. (b) Ground truth. (c) Final output of the last negative attention shifting block. (d) Output of the main branch. (e) Output of the attention branch, attention mask. (f) Output of the main branch after multiplication with the attention mask from the attention branch. (g) Differences on the output before and after the attention mask is applied. As shown in (g), for example in the zoom-in region, negative changes (blue areas) can be observed on the foreground.
Figure 8: Results of expected calibration error (ECE) against segmentation performances (IoU) from cross-validation on CARVE with 50 labelled slices. ECE is calculated using accuracy and confidence: \( ECE = \sum_{m=1}^{M} \frac{|B_m|}{n} |acc(B_m) - conf(B_m)| \). The lower the ECE value, the better the model is calibrated. In the above plot, each data point represents a pair of ECE and IoU of one testing image. This plot shows that MisMatch learns more trustworthy representations thanks to the consistency regularisation on the unlabelled data. We also observe that a lot of samples enjoy both low calibration error and high segmentation accuracy at the same time, especially among the testing samples from our method. This aforementioned phenomenon might imply that our method improves the calibration leading to better segmentation results.
Figure 9: More reliability maps. Blue: Confidence. Red: Accuracy. Each row is on one testing image. X-axis: bins of prediction confidences. Y-axis: accuracy. Column 1: Sup1 (supervised learning with U-net). Column 2: outputs of positive attention decoders. Column 3: outputs of negative attention decoders. Column 4: average outputs of the two decoders. The smaller the gap between the accuracy and the confidence, the better the network is calibrated.