Note:nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation

Sinkoo

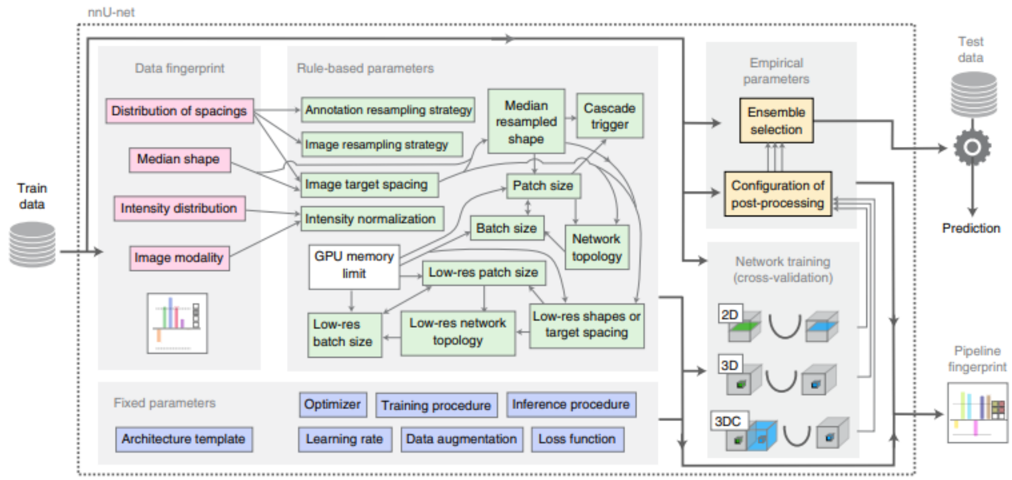
March 27, 2022

1 Abstract

While semantic segmentation algorithms enable image analysis and quantification in many applications, the design of respective specialized solutions is non-trivial and highly dependent on dataset properties and hardware conditions. So author developed nnU-net that automatically configure itself, and surpasses most of existing approaches.

2 Model

Main recipe:
1. Collect design decisions that do not require adaptation between datasets and identify a robust common confguration ('fxed parameters').
2. For as many of the remaining decisions as possible, formulate explicit dependencies between specifc dataset properties ('dataset fngerprint') and design choices ('pipeline fngerprint') in the form of heuristic rules to allow for almost-instant adaptation on application ('rule-based parameters').
3. Learn only the remaining decisions empirically from the data ('empirical parameters').



Design choice	Required input	Automated (fixed, rule-based or empirical) configuration derived by distilling expert knowledge (more details in online methods)
Learning rate	–	Poly learning rate schedule (initial, 0.01)
Loss function	–	Dice and cross-entropy
Architecture template	–	Encoder–decoder with skip-connection ('U-Net-like') and instance normalization, leaky ReLU, deep super-vision (topology-adapted in inferred parameters)
Optimizer	–	SGD with Nesterov momentum (μ = 0.99)
Data augmentation	–	Rotations, scaling, Gaussian noise, Gaussian blur, brightness, contrast, simulation of low resolution, gamma correction and mirroring
Training procedure	–	1,000 epochs × 250 minibatches, foreground oversampling
Inference procedure	–	Sliding window with half-patch size overlap, Gaussian patch center weighting
Intensity normalization	Modality, intensity distribution	If CT, global dataset percentile clipping & z score with global foreground mean and s.d. Otherwise, z score with per image mean and s.d.
Image resampling strategy	Distribution of spacings	If anisotropic, in-plane with third-order spline, out-of-plane with nearest neighbor Otherwise, third-order spline
Annotation resampling strategy	Distribution of spacings	Convert to one-hot encoding → If anisotropic, in-plane with linear interpolation, out-of-plane with nearest neighbor Otherwise, linear interpolation
Image target spacing	Distribution of spacings	If anisotropic, lowest resolution axis tenth percentile, other axes median. Otherwise, median spacing for each axis. (computed based on spacings found in training cases)
Network topology, patch size, batch size	Median resampled shape, target spacing, GPU memory limit	Initialize the patch size to median image shape and iteratively reduce it while adapting the network topology accordingly until the network can be trained with a batch size of at least 2 given GPU memory constraints. for details see online methods.
Trigger of 3D U-Net cascade	Median resampled image size, patch size	Yes, if patch size of the 3D full resolution U-Net covers less than 12.5% of the median resampled image shape
Configuration of low-resolution 3D U-Net	Low-res target spacing or image shapes, GPU memory limit	Iteratively increase target spacing while reconfiguring patch size, network topology and batch size (as described above) until the configured patch size covers 25% of the median image shape. For details, see online methods.
Configuration of post-processing	Full set of training data and annotations	Treating all foreground classes as one; does all-but-largest-component-suppression increase cross-validation performance? Yes, apply; reiterate for individual classes No, do not apply; reiterate for individual foreground classes
Ensemble selection	Full set of training data and annotations	From 2D U-Net, 3D U-Net or 3D cascade, choose the best model (or combination of two) according to cross-validation performance

Fig. 2 | Proposed automated method configuration for deep learning-based biomedical image segmentation. Given a new segmentation task, dataset properties are extracted in the form of a 'dataset fingerprint' (pink). A set of heuristic rules models parameter interdependencies (shown as thin arrows) and operates on this fingerprint to infer the data-dependent 'rule-based parameters' (green) of the pipeline. These are complemented by 'fixed parameters' (blue), which are predefined and do not require adaptation. Up to three configurations are trained in a five-fold cross-validation. Finally, nnU-Net automatically performs empirical selection of the optimal ensemble of these models and determines whether post-processing is required ('empirical parameters', yellow). The table on the bottom shows explicit values as well as summarized rule formulations of all configured parameters. Res., resolution.

nnU-Net's automated configuration covers the entire segmentation pipeline without any manual decisions.
nnU-Net generates three different U-Net configurations: a two-dimensional (2D)

U-Net, a 3D U-Net that operates at full image resolution and a 3D U-Net cascade in which the first U-Net operates on downsampled images, and the second is trained to refine the segmentation maps created by the former at full resolution. Empirically chooses the best configuration or ensemble after cross-validation Finally, nnU-Net empirically opts for 'non-largest component suppression'(Like NMS?) as a post-processing step.

3 Summary

nnU-Net('no new net') underscores the relative importance of method configuration over architectural variations. The strong performance of nnU-Net is not achieved by a new network architecture, loss function or training scheme, but by systematizing the complex process of method configuration.Giving the idea of thinking about method configuration instead of concentrating on network architecture when encounter a problem in deep learning based segmentation sometimes.