Recognition of tomographic images in the diagnosis of stroke

Kalmutskiy Kirill

April 2020

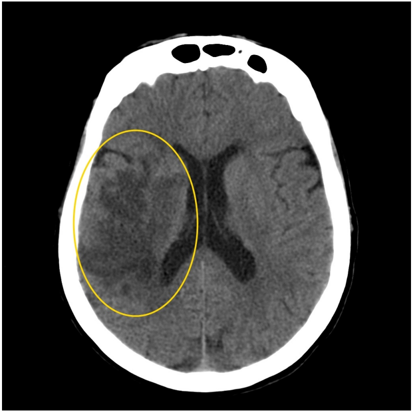
Formulation of the problem

A large dataset of tomographic images of the brain is given. Each image is a section of the brain. Manual marking of such a dataset is very expensive, since only a highly qualified specialist can do this.

In this regard, it is necessary to create a model that will predict areas of stroke and select them on tomographic images of the brain.

The complexity of the task is that there are very few labeled images.

Initial data



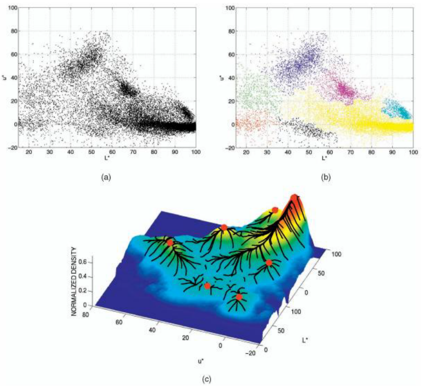
- Images in DICOM format;
- $\sim$30 labeled images of brain slices;
- More than 30 GB of unlabeled images;

Segmentation task

Image segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as image objects).
More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain characteristics.

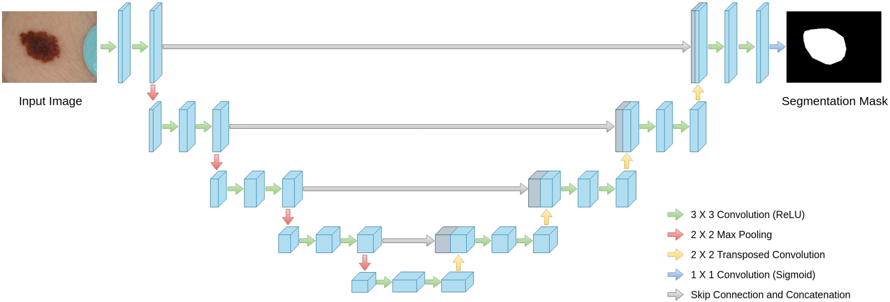
- Automatic segmentation;
- Semi-automatic segmentation;
- Trainable segmentation

Automatic segmentation



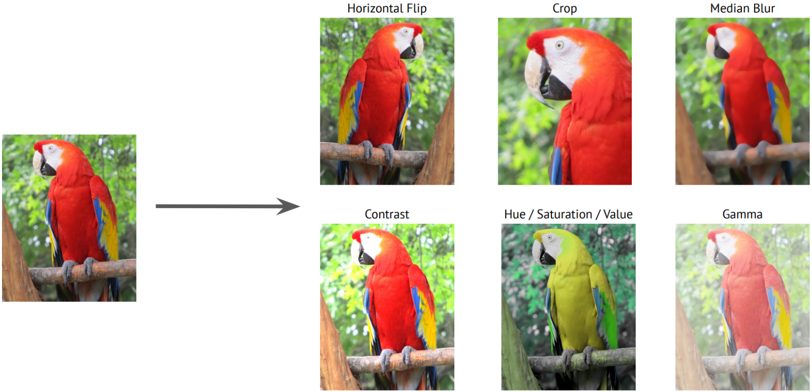
- WaterShed - good if the image has few local minima;
- MeanShift - good if the desired object is almost uniform in color;

Trainable segmentation

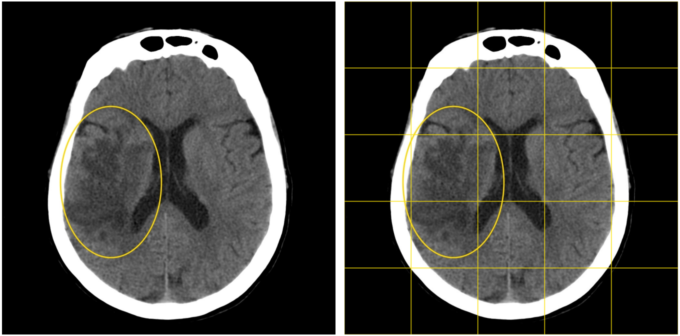


- The model allows you to achieve high metrics;
- Pre-trained models can be used;
- A lot of data is needed for training;
- Inability to interpret model decisions;

Augmentation



Cropping into puzzle



- Learning on crop smaller than the original image;
- To predict, we cut the picture into crop again, predict for them and collect the image back;
- We can increase the number of images up to 36 times;

Interpretation of Results

Unfortunately, neural network results cannot be directly interpreted. However, well-interpreted algorithms can be used for these purposes. For example, decision tree based on similarity.

Decision trees based on similarity is a binary decision tree that uses a measure of object similarity to key points instead of predicates.

Cluster centers can be used as key points, and the distance to these points - as a measure of similarity.

We can train this classifier on the features obtained at the output of the encoder.

Task specifics

		Actual	
		Positive	Negative
Predicted	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

- The quality of segmentation (i.e., IoU) is not so important;
- Accuracy and the absence of errors of the first kind (False Positive) are important (sick patients should not be missed);
- At the same time, a reasonable number of mistakes of the second kind (False Negative) are allowed (however, doctors should not be very irritated, otherwise they simply will not use the model).

Segmentator and classifier

There is a way how to increase accuracy

- Add linear layer after encoder output layer;
- Use this layer as a classifier;
- Train the model on the sum of the classifier and segmentator losses;
- For prediction: segmentation is performed only if the classifier found a stroke in the image;

Putting it all together

- Model: U-Net-like neural network;
- Extension dataset: Augmentation and Cropping into puzzle;
- Interpretability: Decision tree based on similarity;
- Accuracy increase: Use of classifier layer after encoder;

Sources

- Olaf Ronneberger "U-Net: Convolutional Networks for Biomedical Image Segmentation", https://arxiv.org/abs/1505.04597
- Dingding Liu: "A Review of Computer Vision Segmentation Algorithms", https://courses.cs.washington.edu/courses/cse576/12sp/notes/remote.p
- "InceptionV3 (prev. ResNet50) Keras baseline model", https://www.kaggle.com/akensert/inceptionv3-prev-resnet50-keras-baseline-model
- Anton Kornilov: "An Overview of Watershed Algorithm Implementations in Open Source Libraries", https://www.researchgate.net/publication/328468060_An_Overview_of_W
- Connor Shorten: "A survey on Image Data Augmentation for Deep Learning", https://link.springer.com/article/10.1186/s40537-019-0197-0
- Adrian Rosebrock: "Intersection over Union (IoU) for object detection", https://www.pyimagesearch.com/2016/11/07/intersection-over-