HAL
open science

Comparison of CNN architectures and training strategies for quantitative analysis of idiopathic interstitial pneumonia

Simon Rennotte, Pierre-Yves Brillet, Catalin Fetita

To cite this version:

Simon Rennotte, Pierre-Yves Brillet, Catalin Fetita. Comparison of CNN architectures and training strategies for quantitative analysis of idiopathic interstitial pneumonia. MEDICAL IMAGING 2020: Computer-Aided Diagnosis, Feb 2020, Houston, United States. pp.113140B:1-113140B:10, 10.1117/12.2548476 . hal-03127665

HAL Id: hal-03127665

https://hal.science/hal-03127665v1

Submitted on 15 Mar 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Comparison of CNN architectures and training strategies for quantitative analysis of idiopathic interstitial pneumonia

Simon Rennotte[1], Pierre-Yves Brillet[2,3] and Catalin Fetita[1]

[1]Samovar, Télécom SudParis, Institut Polytechnique de Paris, France
[2]Université Paris13, Paris, France
[3]Avicenne Hospital, AP-HP, Bobigny, France

Abstract. Fibrosing idiopathic interstitial pneumonia (IIP) is a subclass of interstitial lung diseases manifesting as progressive worsening of lung function. Such degradation is a continuous and irreversible process which requires quantitative follow-up of patients to assess the pathology occurrence and extent in the lung. The development of automated CAD tools for such purpose is oriented today towards machine learning approaches and in particular convolutional neural networks. The difficulty remains in the choice of the network architecture that best fit to the problem, in straight relationship with available databases for training. We follow-up our work on lung texture analysis and investigate different CNN architectures and training strategies in the context of a limited database, with high class imbalance and subjective and partial annotations. We show that increased performances are achieved using an end-to-end architecture versus patch-based, but also that naive implementation in the former case should be avoided. The proposed solution is able to leverage global information in the scan and shows a high improvement in the F1 scores of the predicted classes and visual results of predictions in better accordance with the radiologist expectations.

Keywords: infiltrative lung diseases, fibrosing idiopathic interstitial pneumonia, lung texture classification, convolutional neural networks, deep learning, locally connected filters, UNet

1. Introduction

Interstitial lung diseases (ILD) include more than 200 chronic lung disorders. Among them, fibrosing idiopathic interstitial pneumonia (IIP), whose cause is mostly unknown, lead to fibrosis in a continuous and irreversible process of lung function decay [1]. The IIP diagnosis, follow-up and treatment adjustment is largely based on texture analysis of pulmonary lesions depicted with computed tomography scan (CT-scan) [2-5]. The development of automated CAD tools for such purpose is oriented today towards machine learning approaches and in particular convolutional neural networks. The difficulty remains in the choice of the network architecture that best fit to the problem, in straight relationship with available databases for training.

In a previous work [6], we have addressed the problem of lung texture classification in ILD by using a patch-based CNN architecture to predict three classes in the lung: normal tissue, IIP (fibrosis+ground glass) and emphysema. However, this network presents several inherent limitations. First, the patches must be prepared before each inference. The image must be preprocessed to attenuate vascular opacities for higher precision in detecting IIPs [6], the lungs must be pre-segmented for definition of training patches, and for each pixel inside the lung, the corresponding patch must be extracted. Only then the network must perform inference on every patch. This procedure limits the usage of the network in clinical (near-)real time applications. Additionally, the performances of the network are poor for patches where the local information is insufficient to distinguish between classes. For example, this led the patch network to classify some vessels, that were not well attenuated, as fibrosis.

For these reasons, in this study of IIPs, we shall extend our investigation to an end-to-end architecture, which considers the whole image as input, and can thus exploit both local and global information in the image. Due to the difficult nature of the available database, we show that the training with the end-to-end architecture needs the usage of several techniques to correctly predict the less represented classes. Different methods to deal with this database specificities (which can be encountered in most of individual hospital databases) will be presented and evaluated. In this paper, we shall present the results obtained with different learning strategies using the end-to-end network and then compare the best version with the network using patches, in terms of test metrics and visual analysis of the results in particular cases. We finally discuss preliminary results obtained by applying the best network configuration to 3D quantification and follow-up of IIPs.

2. Materials and methods

2.1 The database

The database of this study was collected at Avicenne Hospital, Bobigny, France (Paris area) which is a French reference center for infiltrative lung diseases. The database includes 156 patients totaling 2266 axial images. Among them, 137 patients (2076 slices) are used for train and validation and 19 patients (190 slices) for test. The ground truth annotations were performed manually by an expert radiologist using an in-house software. Hence, the training data is very expensive to create, which might be a limitation for network generalization in clinical practice in case of a reduced training database size, especially for end-to-end networks where an input sample is the whole image (contrary to patch-based networks). This led us to select the UNet architecture [7] in our study, because of its great success in medical applications, for which we kept a relatively small number of parameters with respect to other end-to-end architectures, in order to reduce the overfitting risk with this training database.

The second problem of our database is the large imbalance between the classes. For the training data, the distribution of each class is as follows: normal 12.3 %, emphysema 0.3 %, IIP 3.8 %, non-lung: 83.6 %. This will induce (as shown later) that an implementation of an end-to-end neural network using the classic cross-entropy loss may underestimate the less representative classes (IIP and emphysema) and overestimate the non-lung and normal classes.

The third issue comes from the highly variable occurrence of different classes (especially emphysema and IIP) from patient to patient. Some patients do not have any emphysema, and the proportion of emphysema varies greatly. Hence, when the database is split between training and validation sets in a naive random way, the proportion of emphysema and fibrosis in the training set and validation set can be very different from one test to another. This can lead to cases where there are not enough emphysema samples in the training split, and the model cannot learn to predict this class. In practice, this problem results in a large standard deviation of the accuracy of the emphysema and fibrosis class for different tests with the same parameters. This makes the comparison of the results obtained with different architectures or techniques very difficult since the metrics variation for the same parameters is higher than the variations due to architecture or optimization changes.

Finally, there is an inherent subjectivity in the annotation of the database, both intra- and inter-expert. It can be hypothesized that this have an adverse effect on the capacity of the neural network to learn. For example, the emphysema class consists of air regions in the lung. However, some regions can be too small and numerous for an accurate manual delineation. In practice, sometimes, the lines for separation are drawn over multiple regions and sometimes some regions are not annotated. This can explain why the emphysema class is difficult to predict by the network, even if a simple intensity threshold can detect most of it (for non-ambiguous situations). Different examples of ground truth annotations illustrating the above-mentioned flaws are shown in Figure 1.

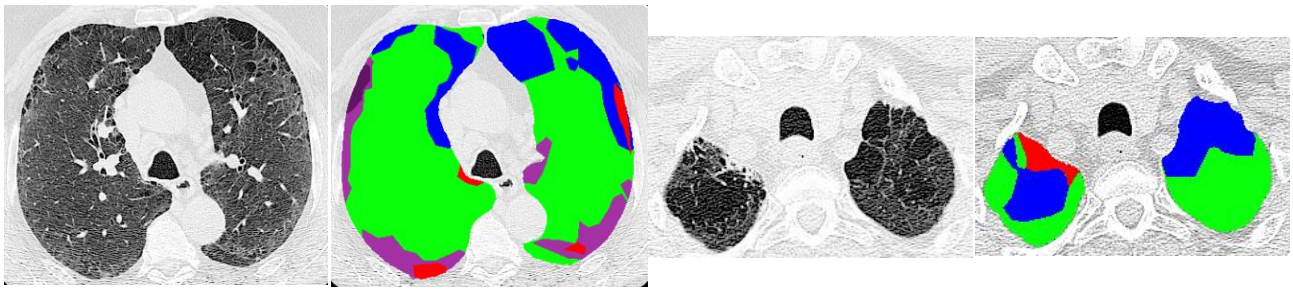


Fig 1: Examples of partial and subjective ground truth (green = lung, blue= emphysema, red = ground glass, purple = fibrosis).

In the following sections, we shall present the techniques used to overcome these problems in the case of patch-based and end-to-end model architectures.

2.2 Preprocessing

Because of the similar appearance of vessels and IIP reticulations in the lung images, we might expect that neural networks will have difficulties for correct prediction of some normal and IIP regions. A solution consists of “removing” vascular structures in the image by attenuating their intensity. Such preprocessing technique is described in [6] where it was shown that its use results in better accuracy for the patch-based network. Moreover, the lungs and airways were automatically segmented to provide an analysis region for the patch-based network.

Before the network training, the separation between the training and validation sets is performed. An iterative algorithm is used to select patients for training and validation to obtain comparable classes proportion between training and validation set slices. This method reduces the variability problem explained in §2.1.

2.3 Network architecture with patch input

The DT-CNN architecture used in [6] was considered in this study. It consists of 8 convolutional layers (2x2 kernels) with Leaky RELU, followed by 4 fully-connected layers. The input patches of 16x16 pixels are extracted from image regions falling inside the lung mask. For a patch to be assigned a given class, the class of the central pixel must be present in 80% of the pixels in the patch. Three target classes were considered: normal, emphysema and IIP. There is thus the possibility of selecting a large number of training patches for each class, which limits the effect of class imbalance in the training data. To further reduce this problem, patches for training are selected so that the same number of patches for each class is used in each epoch. At each epoch, the procedure is repeated, and different patches are selected. A coefficient was also added to allow finetuning the proportion of patches of each class. In practice, retaining the same number of patches for each class leads to an overestimation of the class less present in the first place. This hyper parameter can be changed to reduce the overestimation of a given class, if needed. Note that the preprocessing step (consisting of vessel "removal") must always be performed, otherwise the performances decline very sharply [6]. This results in a very high processing time by slice, preventing the use of patch-based networks for real time applications.

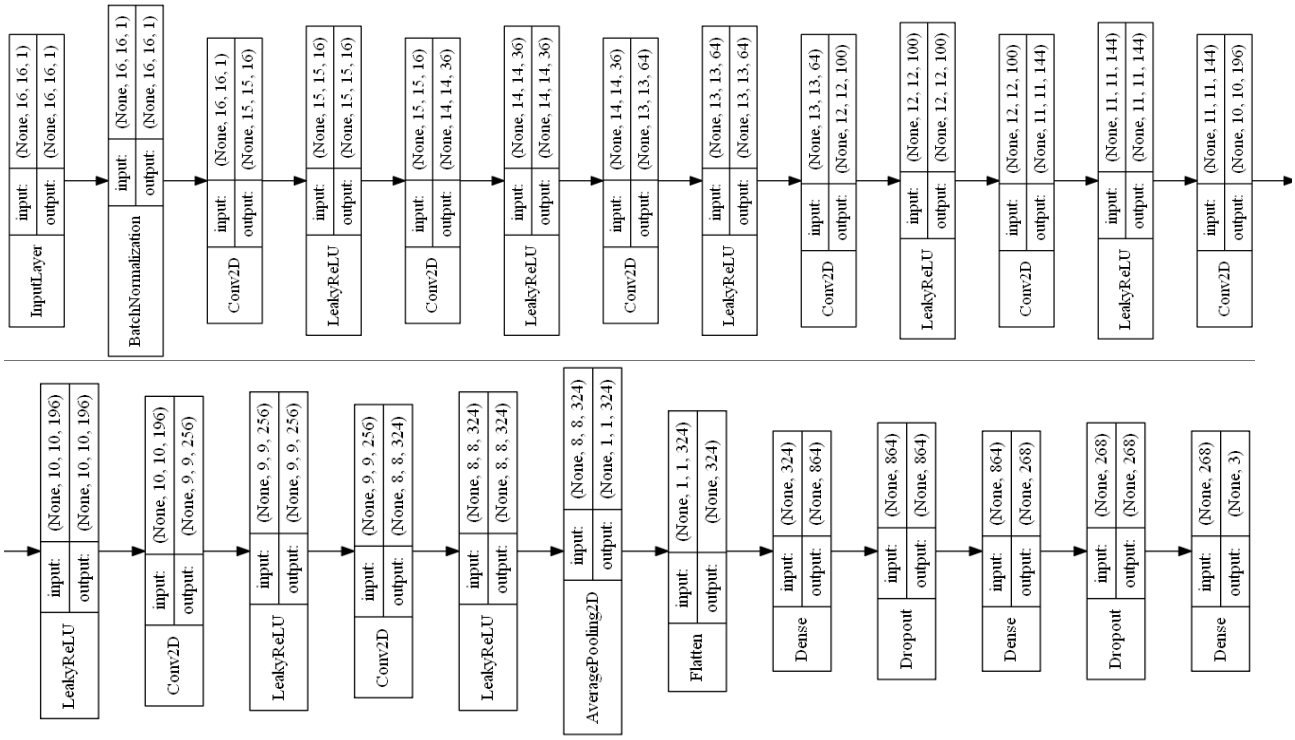


Fig. 2. Network architecture for the DT-CNN [6].

2.4 End-to-end network architecture: UNet

In parallel, we investigated here a network that takes as input a whole image to output the segmentation map of the three classes. As mentioned before, UNet [7] was here selected, due to its wide success in medical segmentation applications, for which we reduced the number of parameters (Fig. 3). A batch normalization block is included at the network input to avoid particular data normalization. A fourth class, *non-lung*, is added to allow the network learn to segment the lung and thus reduce the preprocessing step requiring lung segmentation in §2.3. This may result in some rare cases where the segmentation of the lungs show some issues, but in general, it could allow to avoid the preprocessing step, if no vascular prefiltering is required.

An aspect of interest is thus to validate the necessity of this vascular prefiltering. Since the prediction here uses the global information in the image in a multi-resolution approach, we might expect compensating the negative impact of using the original data instead of vascular-filtered data, which will highly accelerate the analysis.

Moreover, intuitively, a better prediction is expected if three-dimensional information is considered in input. In this respect, we analyzed two network configurations: the first one is the classic architecture with 2D image input; the second one considers 11 adjacent axial images in the CT volume as input. From our experimental tests, it turned out that the

network using the 3D information overfitted, probably because of insufficient samples in the available database.

We thus concentrated on the classic 2D architecture and optimized several parameters to overcome the database issues discussed in §2.1. Two aspects were investigated here, the learning optimization with respect to the loss function, and the use of vascular prefiltering. This analysis is presented in the following subsections.

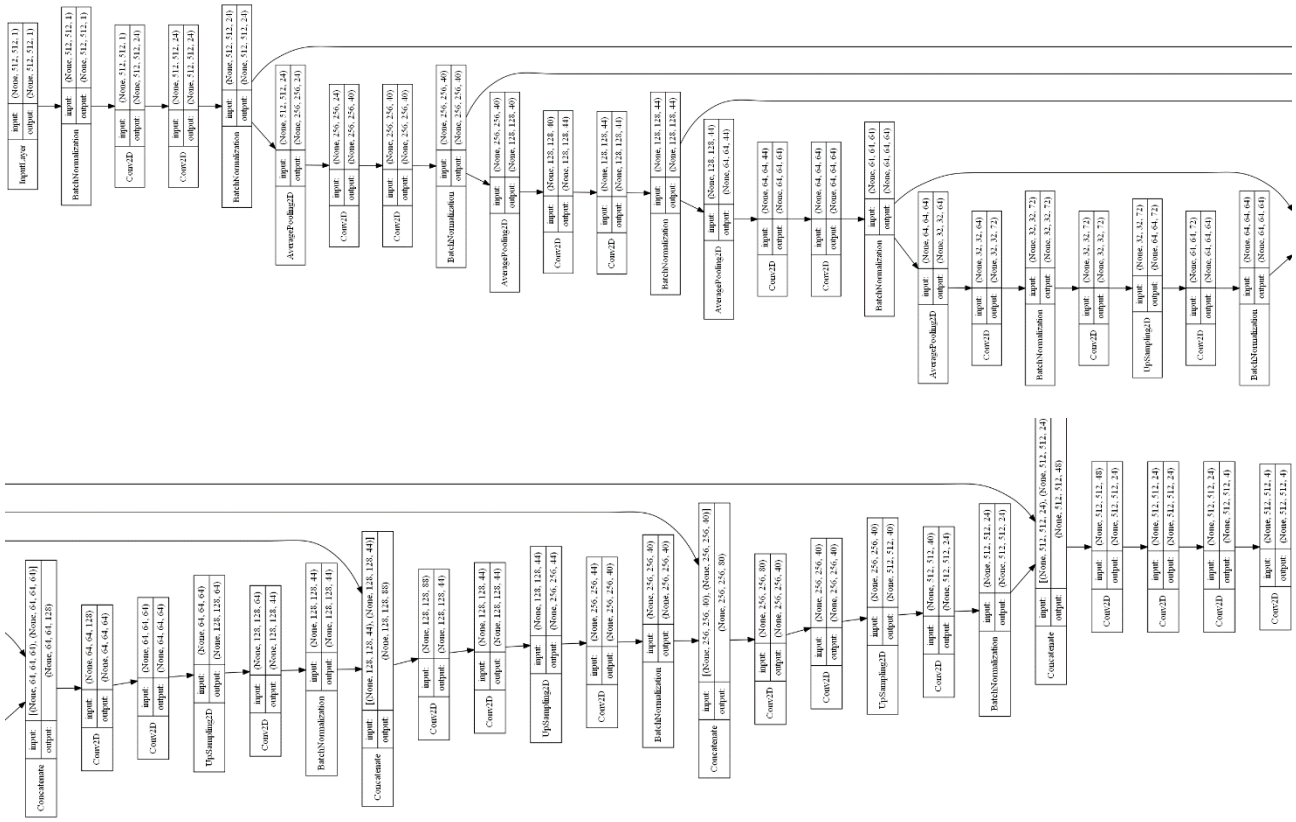


Fig. 3. UNet architecture [7] used in this study, with less number of convolution filters at each level of resolution (24 : 40 : 44 : 64 : 72 : 64 : 44 : 40 : 24) and a batch normalization block at network input.

Data augmentations methods were used to limit the overfitting and improve the generalization of the network to different body sizes, different CT reconstruction filters and different body mass indexes (BMI), which impact the mean pixel value inside the lung. The python library *Albumentations* [8] was applied to shift, scale and rotate the images randomly, as well as changing the luminosity, the contrast, and perform random masking of the image by replacing pixels in a square by a value of 1000 (Hounsfield Units).

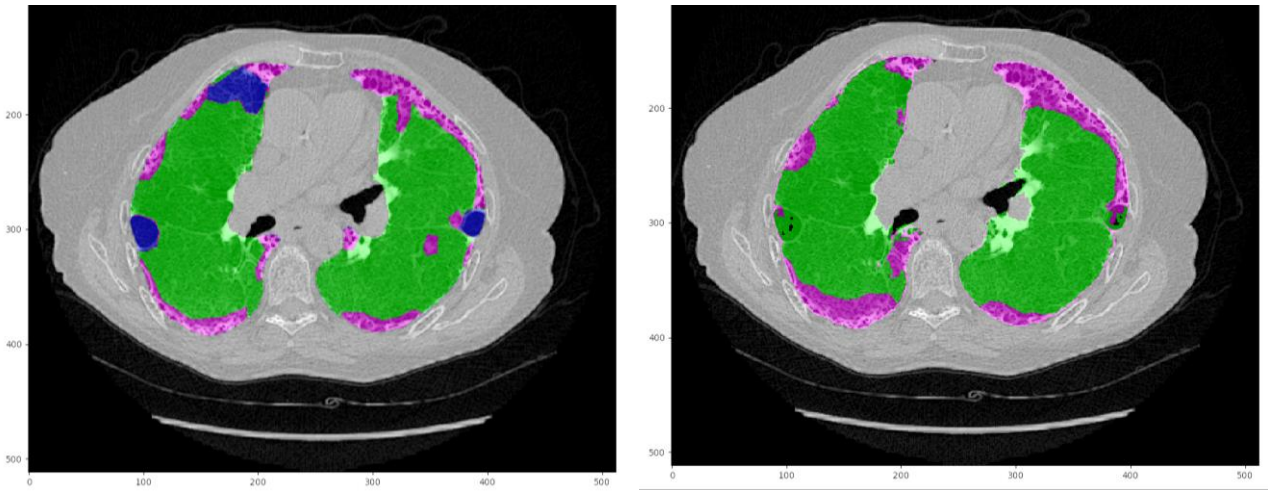
2.4.1 Loss selection

Several losses were investigated for the end-to-end network: classic cross-entropy, weighted cross-entropy and Dice loss.

Cross-entropy loss

For the available database, the UNet training using the cross-entropy loss function (eq. 1) failed to predict the emphysema class, due to the class imbalance problem (Fig. 4).

$$\text{CE } loss(y_{true}, y_{pred}) = -\frac{1}{N_{pixels}} \sum_{pixel}^{Npixels} \mathrm{y}_{\mathrm{true}_{\mathrm{pixel}}} \cdot \log\left(y_{pred_{pixel}}\right) \quad (1)$$



(a) Ground truth (b) UNet prediction using cross-entropy loss

Fig 4. Illustration of error predicting emphysema using UNet with cross-entropy loss. Color coding: green=normal, pink=IIP, blue=emphysema.

In the patch-based network case, the class imbalance problem was solved by choosing an equivalent number of patches of each class in each batch during the training. In the end-to-end network case, this is no longer possible since the whole image is used as input, and the *non-lung* class by far, followed by the *normal* class are more present in the labels. This problem has been solved for UNet by choosing a loss function robust to the class imbalance problem. We investigated further modifications of the loss function in order to reduce the impact of the label noise on the results.

Weighted cross-entropy loss

In order to correct the relative importance of each class during training, the first option was to use a weighted cross-entropy loss. A multiplicative term that depends on the label of the pixel is added to the cross-entropy loss, as shown in the eq. 2:

$$WCE\ loss(y_{true}, y_{pred}) = -\frac{1}{N_{pixels}} \sum_{pixel}^{Npixels} \mathrm{w}(\mathrm{y_{true}}) \cdot \mathrm{y_{true_{pixel}}} \cdot \log(y_{pred_{pixel}}) \quad (2)$$

In this way, it is possible to compensate the lack of emphysema pixels by attributing a higher term to the emphysema class. The selection of *w* value for each class is not straightforward. A first approach would be attributing a value to each class, which is inversely proportional to the class extent in the training images, and then normalize *w* so that the sum of the values for each class equals 1:

$$w(class) = normalized(1/N_{class}), \quad (3)$$

where N_{class} denotes the number of class pixels in the training database.

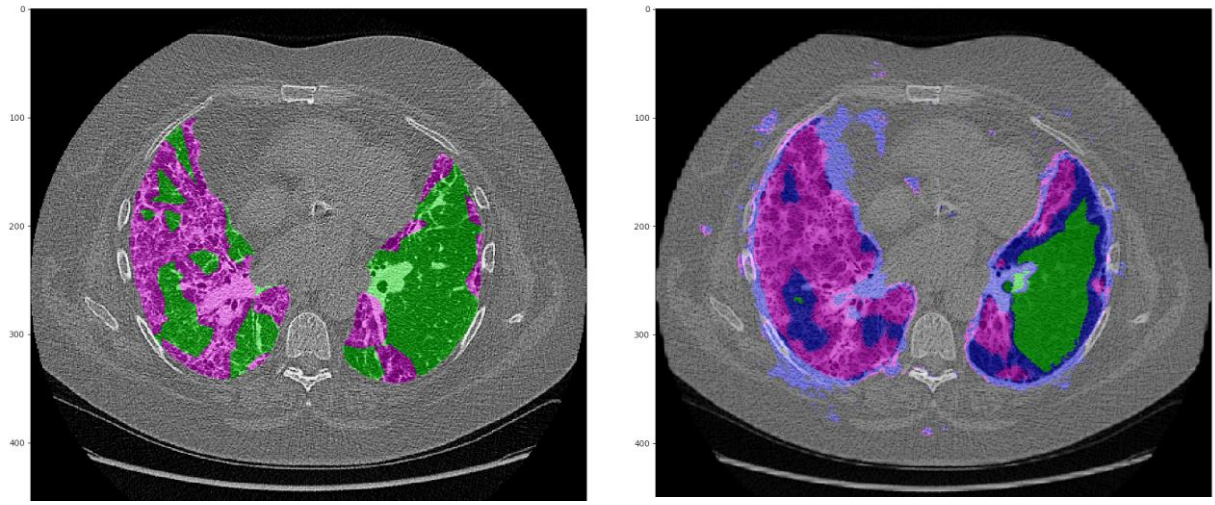
We thus obtain w(N)=0.026; w(E)=0.879; w(IIP)=0.092; w(nL)= 0.003, N denoting the normal class, E – emphysema and nL – non-lung. As expected from the *w* values obtained above, this choice of *w* tended to overestimate the emphysema class (Fig. 5).

The second approach considered to select the values of *w* was to apply a logarithm to the values obtained by the first method:

$$w(class) = normalized(1/log(N_{class}/min(N_{class})+1)). \quad (4)$$

In this case we obtained a better distribution of the class weights, w(N)=0.120; w(E)=0.620; w(IIP)=0.182; w(nL)= 0.078, but still an overestimation of the IIP versus *normal* class was noticed, Fig. 6a. In order to find the appropriate balance between IIP and N class, we introduced a fine-tuning coefficient α to reassign the value of *w*(IIP) based on the initial values of *w* for IIP and N computed from eq. 4, as follows:

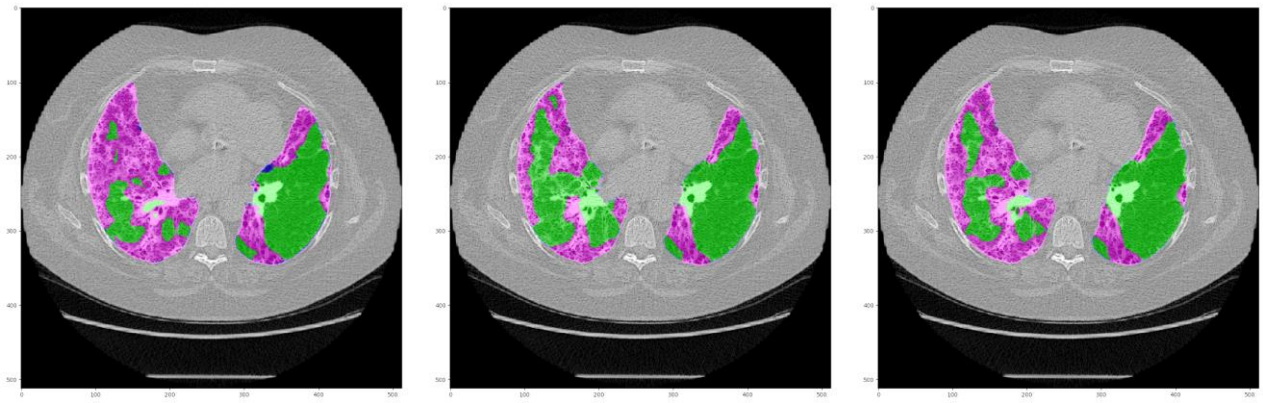
$$w(IIP) = \alpha \cdot (w(IIP) - w(N)) + w(N). \quad (5)$$



(a) Ground truth (b) UNet prediction using linear weighted cross-entropy loss

Fig. 5. Illustration of emphysema overestimation using UNet with linear weighted cross-entropy loss. Color coding: green=normal, pink=IIP, blue=emphysema.

Note that α=1 does not modify the initial values for *w*(IIP) and *w*(N) leading to the overestimation of the IIP class (Fig. 6a), while α=0, leading to identical weights for *w*(IIP) and *w*(N), (w(N)=0.129; w(E)=0.6260; w(IIP)=0.129; w(nL)= 0.082) will result in underestimating the IIP class, Fig. 6b. Finetuning the α value between 0 and 1 led to the optimal value α=0.2 achieving the best compromise between visual assessment and quantitative scores, Fig. 6c, Table 1.



(a) α=1, IIP overestimation (b) α=0, IIP underestimation (c) α=0.2, best prediction

Fig. 6. Finetuning α value in log-weighted entropy loss, eqs. (4),(5). Same ground truth as in Fig. 5a.

	α=1		α=0		α=0.2	
	precision	**recall**	**precision**	**recall**	**precision**	**recall**
class						
N	0.93	0.88	0.91	0.92	0.91	0.91
E	0.29	0.59	0.24	0.57	0.21	0.37
IIP	0.61	0.77	0.73	0.67	0.69	0.73
no-lung	1.00	0.99	1.00	1.00	1.00	0.99
global accuracy	**0.97304**		**0.97676**		**0.97632**	

Table 1. Quantitative performance for different α values in log-weighted entropy loss, eqs. (4),(5).

Dice loss

An alternative to weighted cross entropy is the use of the Dice loss, which does not introduce new hyperparameters to finetune:

$$Dice\ loss(y_{true}, y_{pred}) = 1 - \frac{1}{N_{class}} \sum_{class=1}^{N_{class}} \frac{2TP_{class}(y_{true}, y_{pred})}{2TP_{class}(y_{true}, y_{pred}) + FP_{class}(y_{true}, y_{pred}) + FN_{class}(y_{true}, y_{pred})}$$

$$TP_{class}(y_{true}, y_{pred}) = \sum_{pixel} y_{true_{pixel}} \cdot y_{pred_{pixel}}$$

$$FP_{class}(y_{true}, y_{pred}) = \sum_{pixel} (1 - y_{true_{pixel}}) \cdot y_{pred_{pixel}} \quad (6)$$

$$FN_{class}(y_{true}, y_{pred}) = \sum_{pixel} y_{true_{pixel}} \cdot (1 - y_{pred_{pixel}})$$

where N_{class} denotes the number of classes taken into account (here N_{class}=4). Note that TP_{class}, FP_{class} and FN_{class} are computed independently for each class prediction, thus the classes have identical weight in the Dice loss function.

The choice of Dice loss led to far better accuracy for emphysema and IIP class than the weighted cross entropy (see §3) and it was further on selected, Fig. 7b. Moreover, in order to consider the possibility of annotation errors in the database, we also tested the Dice loss function modified by using the soft bootstrapping method, as explained in [9]. This was achieved by replacing y_{true} in the Dice loss function (eq. 6) by:

$$y_{true} \leftarrow \beta \cdot y_{true} + (1 - \beta) \cdot y_{pred}, \quad (7)$$

where βwas tested with the values 0.8, 0.9 and 0.95. In our case, no performance improvement was noticed when using the soft bootstraping method (see results in §3).

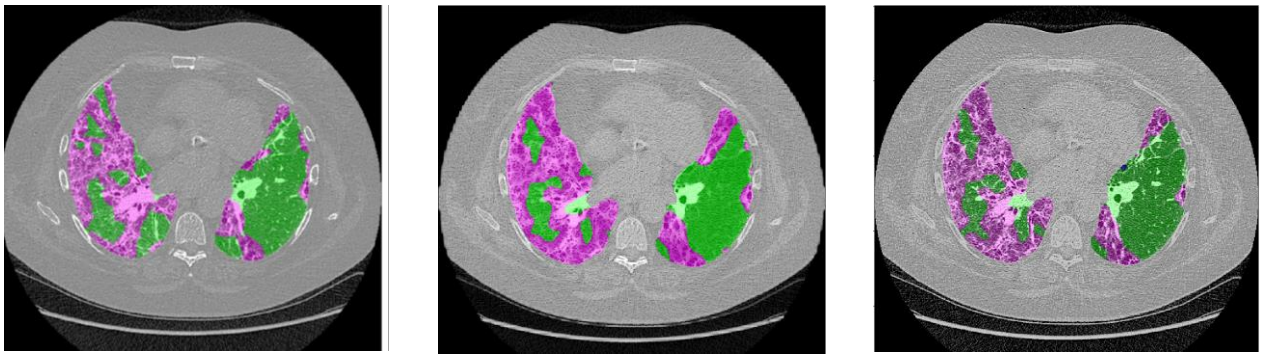
Finally, in order to investigate the possibility of avoiding the data preprocessing step, we have tested the choice of the Dice loss within two network training stages, that is using the native versus vascular-filtered data. This led us to reconsider the optimizer selection as discussed in the following section.

2.4.2 Optimizer selection

In our particular case, a specific local minimum of the loss function turned out to be a problem when the network predicted no emphysema during all the training. This happened for some of the trainings performed with the vascular-filtered images and for all trainings performed on native data (without vascular filtering). In order to fix this problem, instead of using a progressive decay of the learning rate during the training as in previous experiments

$$LR(batch) = 0.925^{\ batch/batch_per_epoch} \cdot LR(0), \quad (8)$$

with LR(0)=0.5, a triangular learning cycle schedule was tested, adapted from [10]. In our case, we selected instead a small period of the cycle during all the training, while varying the minimum value of the momentum (and the maximum value of the learning rate) with time (Fig.8). At the beginning of the training, the maximum learning rate is high, and the minimum momentum is low. This method allowed escaping from the local minimum where the emphysema class was not predicted, and thus obtaining results for the emphysema class in the non-filtered case. The particular learning rate schedule is shown in Figure 8, with the x-axis representing the number of batches. The comparison of the results obtained during the first batches with the progressive decayed learning rate is shown in Figure 9, where the precision for the emphysema class is depicted as fn_1, and the precision for the IIP class as fn_2. We can see that although the performance on the IIP class are equivalent between the two optimization strategies, the emphysema class is not predicted in the progressive decay schedule case. Fig. 7 illustrates a comparison between the prediction of the two networks, with (Fig. 7b) and without (Fig. 7c) vascular prefiltering, showing similar performance.



(a) Ground truth (b) UNet prediction using Dice loss on vascular prefiltered data (c) UNet prediction using Dice loss on native data

Fig. 7. Illustration of UNet prediction with Dice loss and triangular learning cycle schedule. Color coding: green=normal, pink=IIP, blue=emphysema.

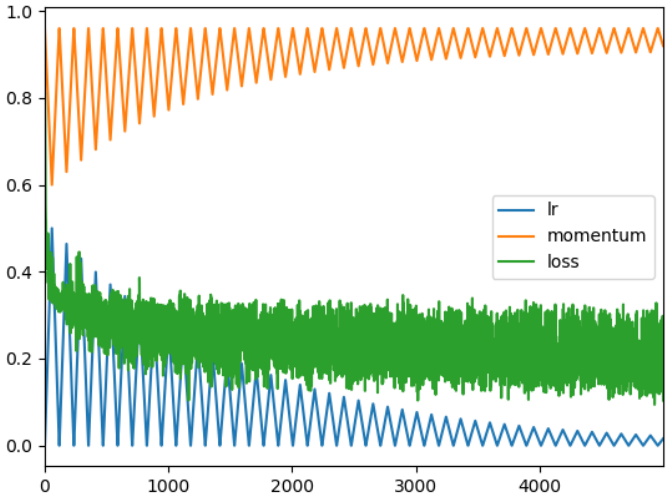


Fig. 8. Learning rate and momentum triangular schedule (batch number plotted in the abscissa).

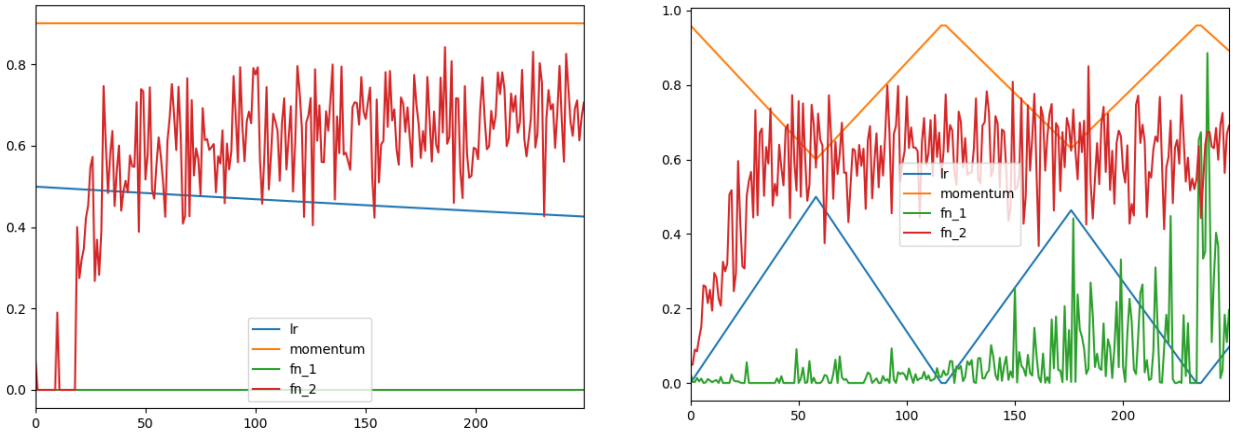


Fig. 9. Comparison of progressive decay learning schedule (left) and a triangular learning schedule (right) in the beginning of training (batch number plotted in the abscissa). Here, fn_1 refers to emphysema, while fn_2 to the IIP class.

3. Results

The results reported in the following on the various comparisons performed in §2 were obtained by using the Keras framework, with Tensorflow as backend. The metrics results for the network configurations discussed in §2.3 and §2.4 are presented in Table 2 in terms of F1-score, computed for each class separately (see eq. 6):

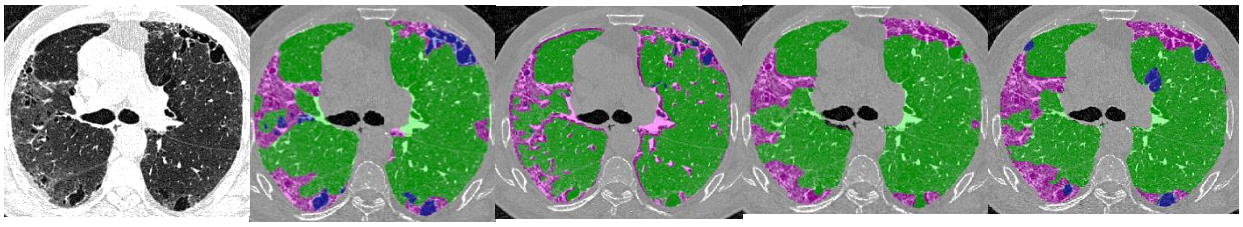
$$F1score_{class}(y_{true}, y_{pred}) = \frac{2TP_{class}(y_{true}, y_{pred})}{2TP_{class}(y_{true}, y_{pred}) + FP_{class}(y_{true}, y_{pred}) + FN_{class}(y_{true}, y_{pred})} \quad (9)$$

All the different losses were tested when using the vascular-filtered data, and the best loss function was retained to perform tests without the vascular prefiltering algorithm. For our test case, the Dice loss performs better than the weighted cross-entropy loss. We can see in Table 2 that the test metrics of the UNet architecture for the classes are better than those obtained with the patch-based network. Note also the improvement due to the data augmentation with the Dice loss. However, the bootstrapping method did not lead to better results. When using a schedule with progressive decay of learning rate, the network did not predict the emphysema class for training and inference performed on native data (no vascular filtering), but the other scores were not altered. When using the triangular learning cycle for the learning rate, all the scores for the non-filtered case were the same as for the filtered case. An example of prediction on a raw CT slice is shown in Figure 10.

Network configuration	F1-score normal	F1-score emphysema	F1-score IIP	F1-score non-lung
Patch-based network	0.83	0.4	0.63	1.00(by default)
UNet with classic cross-entropy loss	0.91	0.00	0.60	1.00
UNet with weighted cross-entropy loss	0.9	0.3	0.69	1.00
UNet with Dice loss	**0.91**	**0.6**	**0.72**	**1.00**
Best UNet with bootstrapping method	0.91	0.58	0.72	1.00
Best UNet wo vascular filtering	0.90	0.00	0.72	1.00
Best UNet wo vascular filtering and triang	**0.91**	**0.6**	**0.72**	**1.00**

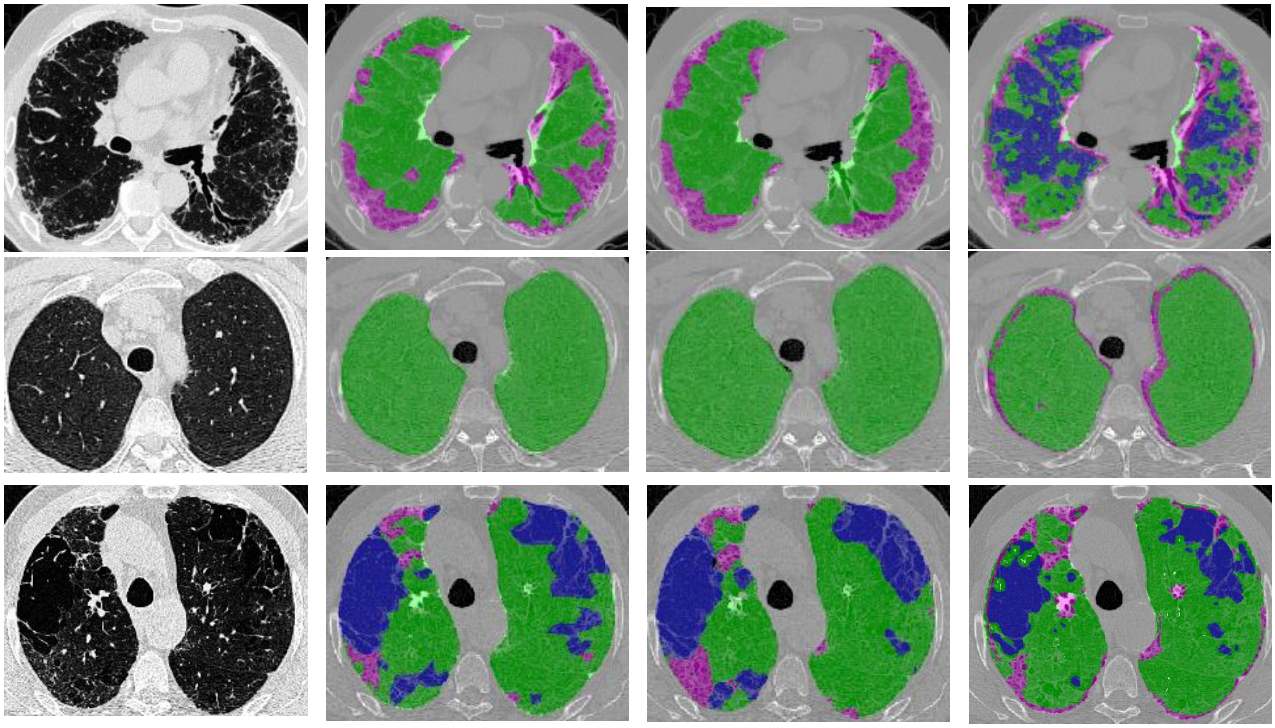
Table 2: Metrics results on the test data for different network configurations and learning parameters.

Some examples of predictions on vascular-filtered data are shown in Figure 11. The first row presents the results obtained on an obese patient. We can see that in this case, the patch network wrongly predicts the emphysema class. Indeed, the pixels values inside the lung are close to the emphysema class and the patch-based network cannot use global information to guide its decisions. The second row shows the results on a normal patient, where again, we can see that the patch-based network finds IIP on the border of the lung, where UNet shows only a minor error. The last images show a typical result for both UNet and the patch-based network. We can see that the predictions are globally correct for UNet. According to the expert remarks, it is difficult to say precisely if the difference between the ground truth and the prediction should be labelled as error, since the line of separation between classes is sometimes controversial.



Original DICOM slice | Ground truth mask | Patch prediction | Best UNet prediction wo triangular schedule | Best UNet w triangular schedule

Fig 10: Examples of predictions on a raw (non-filtered) CT slice. (green = normal, blue= emphysema, pink = IIP).



Original DICOM slice | Ground truth mask | Best UNet prediction | Patch prediction

Fig. 11. Examples of prediction of the UNet and patch-based network on vascular-filtered images. Top row: IIP obese patient, middle row: healthy subject, bottom row: IIP subject (green = normal, blue= emphysema, pink = IIP).

By using the end-to-end network without preprocessing step, the average processing time to predict all the slices of a patient became 10 seconds on a GTX2080Ti, which allowed the possibility to make 3D predictions for a patient in a (near-)real-time scenario.

Once the 2D predictions yielded exploitable results, the network was used to quantify all the slices of the patients in order to have a 3D representation of the patient status. This methodology allows to quickly identify the regions of interests when comparing the same patient at two different follow-up time points, and more easily see the changes. However, this kind of 2D prediction for the 3D representation had some challenges. First, for some patients and some slices, the predictions of the network from slice to slice can be quite different, even if the changes in the image were minor. This resulted in a 3D representation that was not smooth. In order to limit this problem, a post-processing was applied to the prediction result, namely a 3D Gaussian smoothing which locally affects the probability value of each class. The final label of each pixel corresponds to the class with the highest probability. The Gaussian standard deviation ($\sigma_x = \sigma_y = 0.5$; $\sigma_z = 3$) was selected to achieve a tradeoff between a fine axial resolution and smooth changes between slices. Note that the 3D post-processing step did not change the metrics on the test data and, in practice, the predictions were more appreciated by the medical expert. An example of a difference between the predictions with and without post-processing 3D smoothing is shown in Figure 12. On this figure, the predictions on two successive slices are represented, and it can be seen that the 3D smoothing allows predictions to be more spatially consistent.

We may note that, some small segmentations errors could subsist for some patients, especially in the lung bases, or for patient data obtained with a reconstruction filter that was much different from the ones used in our training database. To limit this problem, 3D connected-component analysis is run to retain the two largest components corresponding to the union of the lung classes (N, IIP, E). The class of all other connected components is assigned to *non-lung*. The difference between using the predictions with and without 3D smoothing and connected component analysis is shown in Figure 13.

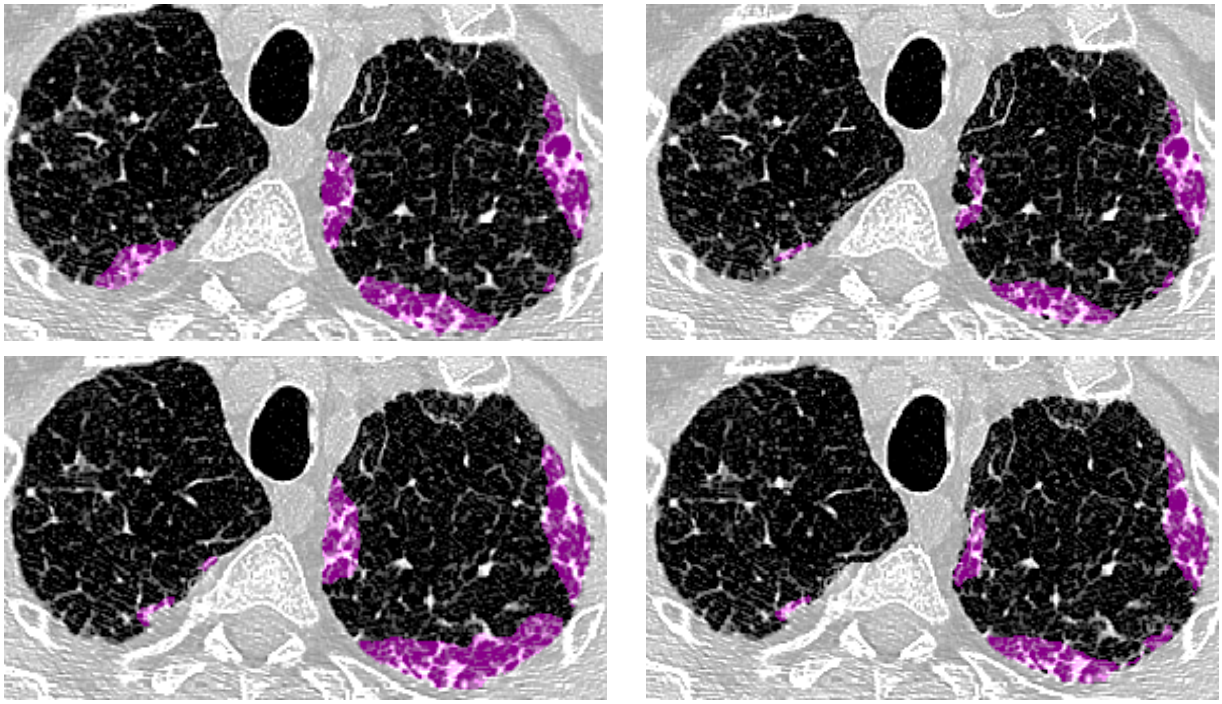
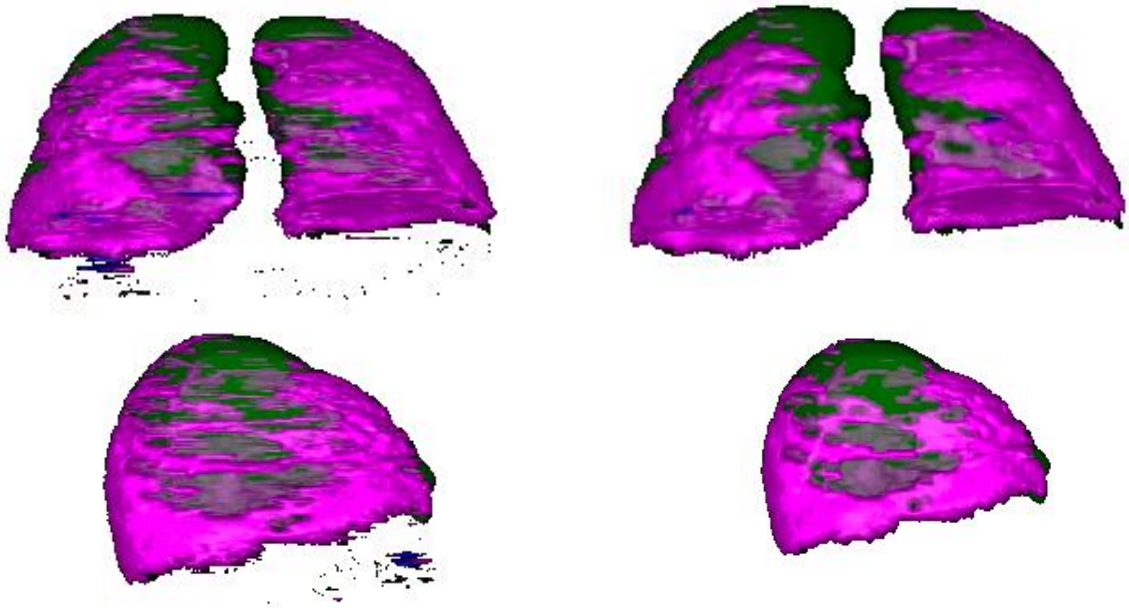


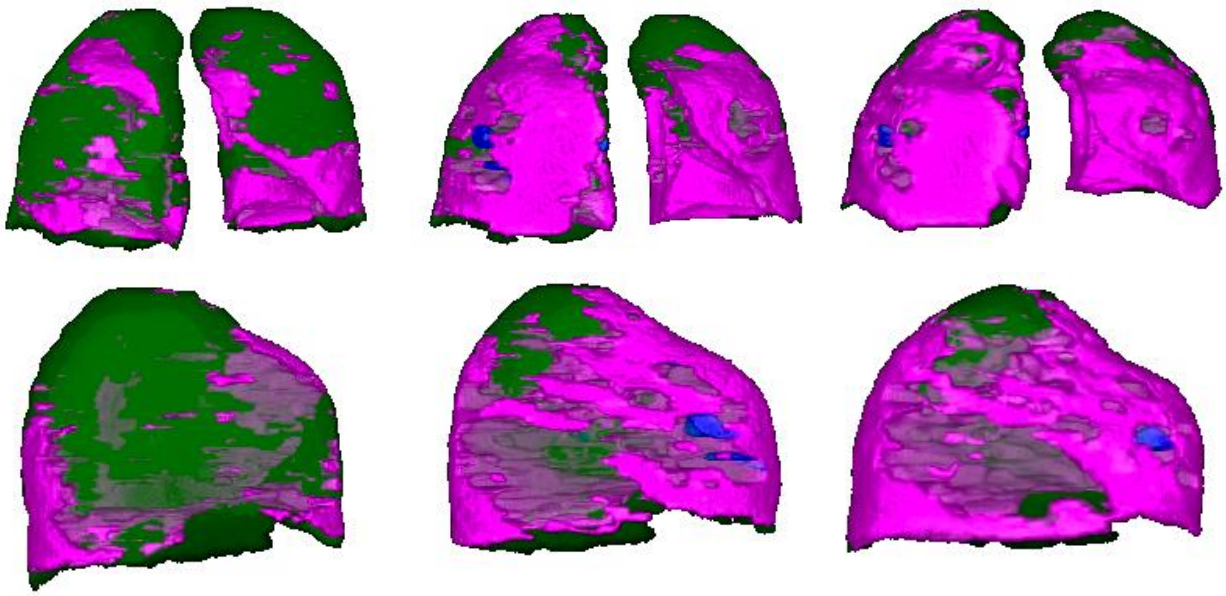
Fig. 12. Successive slice predictions (fibrosis) without (left) and with (right) 3D spatial smoothing.

Finally, the resulting pipeline allows assessing the patient evolution at different follow-up time steps with the use of 3D visualization and the computation of the volume and proportion of each class predicted, as shown in Figure 14.



(a) 3D predictions without postprocessing step (b) 3D predictions with postprocessing step

Fig. 13. Effect of postprocessing step on 3D predictions.



(a) Baseline T0 (b) T0 + 18 months (c) T0 + 36 months

IIP volume (L):	0.16	IIP volume (L):	0.49	IIP volume (L):	0.61
Lung volume (L):	1.32	Lung volume (L):	1.03	Lung volume (L):	1.15
Volume ratio IIP/lung:	12.17%	Volume ratio IIP/lung:	47.62%	Volume ratio IIP/lung:	53.17%

Fig 14. 3D visualizations of the same patient at different time steps and IIP quantitative follow-up (green = normal, blue= emphysema, pink = IIP).

4. Conclusion

This paper proposed a comparison of different CNN configurations for the quantification of interstitial idiopathic pneumonia and discussed the parameter choice in order to tackle problems related to database size and class imbalance.

In this work, we have shown the advantages of end-to-end network compared to network using patches for quantitative analysis of IIP. We have also shown that a naive training with a neural network is sometimes insufficient for databases acquired in local clinical settings. We have used several parameter designs and showed their influence on the prediction results for our specific database. For the best configuration of the end-to-end network, we obtained a significant increase in the F1 score of all classes versus a previous patch-based solution. The resulting pipeline allows performing quantitative follow-up of the IIP patients, with 3D visualization of the disease extent and localization.

References

[1] Raghu G, Collard HR, Egan JJ, Martinez FJ, Behr J, Brown KK, et al. "An Official ATS/ERS/JRS/ALAT Statement: Idiopathic Pulmonary Fibrosis: Evidence-based Guidelines for Diagnosis and Management". *Am J Respir Crit Care Med.* 2011 Mar 15;183(6):788–824.

[2] Lynch DA, Sverzellati N, Travis WD, Brown KK, Colby TV, Galvin JR, et al. "Diagnostic criteria for idiopathic pulmonary fibrosis: a Fleischner Society White Paper". Lancet Respir Med. 2018 Feb;6(2):138–53.

[3] Lynch DA, Godwin JD, Safrin S, Starko KM, Hormel P, Brown KK, et al. "High-Resolution Computed Tomography in Idiopathic Pulmonary Fibrosis: Diagnosis and Prognosis". Am J Respir Crit Care Med. 2005 Aug 15;172(4):488–93.

[4] Sumikawa H, Johkoh T, Colby TV, Ichikado K, Suga M, Taniguchi H, et al. "Computed Tomography Findings in Pathological Usual Interstitial Pneumonia: Relationship to Survival". Am J Respir Crit Care Med. 2008 Feb 15;177(4):433–9.

[5] Bartholmai BJ, Raghunath S, Karwoski RA, Moua T, Rajagopalan S, Maldonado F, et al. "Quantitative Computed Tomography Imaging of Interstitial Lung Diseases". J Thorac Imaging. 2013 Sep;28(5):298–307.

[6] Kim, Young-Wouk & Roberto Tarando, Sebastián & Brillet, Pierre-Yves & Fetita, Catalin. (2019). Image biomarkers for quantitative analysis of idiopathic interstitial pneumonia. 44. 10.1117/12.2511847.

[7] Olaf Ronneberger, Philipp Fischer, Thomas Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation", arXiv:1505.04597

[8] Buslaev A., et al, "Albumentations: fast and flexible image augmentations", arXiv:1809.06839

[9] Reed S., et al, "Training Deep Neural Networks on Noisy Labels with Bootstrapping", arXiv:1412.6596

[10] Leslie N. Smith, "Cyclical Learning Rates for Training Neural Networks", arXiv:1506.01186