Iterated Stacked Classifiers for Lung Segmentation in Computed Tomography

Francesco Ciompi[1,2], Carlo Gatta[2], Marleen de Bruijne[3,4]

1. Dept. of Computer Science, Autonomous University of Barcelona, Spain
2. Computer Vision Center, Campus UAB, Spain
3. Biomedical Imaging Group Rotterdam, Depts. of Radiology and Medical Informatics, Erasmus MC, Rotterdam, The Netherlands
4. Dept. of Computer Science, University of Copenhagen, Copenhagen, Denmark

Abstract. We present a fully automatic method for lung segmentation in Computed Tomography (CT). The method consists in a set of sequentially trained classifiers, each of them fed with information on both pixel appearance and multi-scale analysis of neighbors class likelihood. This approach implicitly embeds contextual information and efficiently encodes long distance interactions. Furthermore, the iteration of classification stages demonstrates to improve the system accuracy. The method is applied to lung segmentation on the public dataset of 55 CT scans presented in the LOLA2011 challenge[1], obtaining an overall score of 0.949.

Key words: Lung segmentation, Computed Tomography, Contextual classification, Stacked Sequential Learning.

1   Introduction

Segmentation of lungs in X-ray Computed Tomography (CT) is an important prerequisite for pulmonary image analysis. It allows the detection and quantification of lung diseases as well as the registration of CT scans.

Segmentation of normal healthy lungs is a rather simple task, due to the low density of lungs in comparison with the surrounding tissues. The presence of severe disease makes the segmentation challenging, although of high clinical interest. As an example, the presence of abnormalities such as tumors, inflammation, or atelectasis, makes the lung to appear as a set of fragmented regions, heterogeneous in density; in some cases, one of the two lungs may not be visible, due to presence of fluid inside the lung. Furthermore, the boundary between the left and right lung is not always clearly visible.

In 2011, a challenge on LObes and Lung Analysis (LOLA2011) was organized, where eight groups participated. Most of the presented methods are based on a *canonical* approach, consisting in alternating several thresholding and region growing steps, based on a set of parameters, often ad-hoc. Variations of canonical methods are presented in [1–6]. In [1], a multi-atlas approach is used in case an erroneous segmentation is detected with the canonical method; in [4] a Cognition Network Language is used as base framework, while the algorithm uses canonical

---

[1] http://www.lola11.com

approach, where information on context, density, geometry and shape is used as well. Such approaches are valid in many of the cases. In [1], authors state that conventional method follows the lung border more precisely than the proposed multi atlas method and the manual tracing when no abnormalities are present. As a matter of fact, the 14 years old method in [2], achieves one of the most accurate result on the LOLA dataset. An approach based on active shape model is proposed in [7], where the model is initialized by detecting ribs, then a robust active shape model is first matched to the data; thereafter, the segmentation is refined using an optimal surface finding algorithm where the search profiles are determined by gradient vector flow. Given the use of a learned model, this approach fails if the lungs shape significantly differs from those of the training set. In [8], a Random Forest classifier is trained in a supervised manner with features related to both lung appearance and context. The method uses the combination of the trees outputs to generate pixel-wise class posterior, and embeds long range visual features by considering the posterior of pixels at a random distance and position with respect to the considered pixel. In order to achieve an accurate lung segmentation, the system is first trained using four classes and then a final label relaxation is applied. This approach is fast and accurate, although it fails in presence of small diameter display field of view images, probably due to the random definition of locations in long range features.

Our contribution. In this paper we present a fully automatic framework for lung segmentation in CT scans. The method consists in an iterative pixel-wise labeling of CT slices, where each stage consists is multi-class classifier. Each classifier is fed with features related to both pixel appearance and context, through a multi-scale analysis of the posterior probabilities of neighbors. In this sense, our approach has some common point with the works presented in [8, 9], although with some important differences. First, differently from [8] our approach is independent on the discriminant classifier used in the framework, where random forest may be an option, but not mandatory. Secondly, differently from [9], contextual information is defined through a rigorous multi-scale approach, which encodes tissue relationships at controlled distances. Each stage of the system is designed following the Stacked Sequential Learning (SSL) principle [10]. We implement the architecture based on [11–13], but we join in a unique framework the multi-scale, iterative and multi-class capabilities: to the best of our knowledge, this is the first time such approach is formulated. We refer to the method as Multi-scale iterative Multi-Class Stacked Sequential Learning (MiM-SSL). The method is tested on a dataset of 55 CT scans of the LOLA2011 lung segmentation challenge. A MATLAB implementation of the code is available upon request to the authors.

2   Method

The method consists of a multi-class classification framework, where a set of stacked classifiers is used. Both *appearance* and *contextual* features are used as
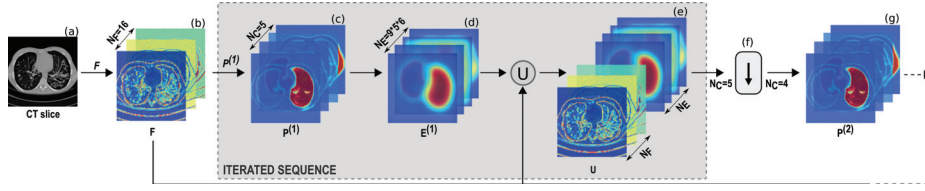
Fig. 1: Schematic representation of the main steps of the framework. N_F features are extracted (b) from the CT slice (a). The pixel-wise probability of each pixel to belong to each class is computed (c), and used as input for the multi-scale analysis (d). Appearance and contextual features are joined (e) to feed the next classification stage. The classes reduction stage (f) is applied only at the first iteration. Additional iterations can be considered in (g).

pixel-wise tissue descriptors. For classes definition, we consider five regions obtained by manual annotation: *left lung*, *right lung*, *air filled spaces inside the body but outside the lung*, *body outside the lung field*, *airspace outside the body*. Since the method is formulated on independent cross-sectional (horizontal) cuts of the CT scan, in the rest of the paper we refer to *axial slice* and to *pixel* as the on-slice component of the voxel.

Multi-class Iterated Multi-scale Stacked Sequential Learning. The proposed lung segmentation method is based on a multi-class iterated extension of the Multi-scale Stacked Sequential Learning (MS-SSL) [11, 12]. The MS-SSL has been proven to be a powerful method for medical image classification [13] and has been iterated in a cascade-like structures for a specific segmentation problem [12].

Here we provide a brief description of the method. Let us define $I(x,y) \in \Re$ (in Hounsfield units) as a slice in the CT scan, with spatial coordinates $\boldsymbol{q} = (x, y) \in \Omega = [1, 2, \ldots, S_x] \times [1, 2, \ldots, S_y]$. When omitting the spatial coordinates, we refer to all the pixels, so that the notation I means the whole image pixels in Ω.

Let us assume that we have a functional $\mathcal{F}$ that maps the input image I in order to extract appearance features, $\mathcal{F} : I \mapsto F \in \Re^{N_F}$, where N_F is the number of features; F shares the same spatial support Ω of I.

Let us also assume that we have a function $\mathcal{P}$ that maps features to label field probabilities, $\mathcal{P} : F(x,y) \mapsto P(x,y) \in [0\,1]^{N_C}$, where N_C is the number of classes to discriminate. This functional can be implemented by means of a classifier or an ensemble of classifiers. During inference, the estimated label field Y can be computed as the maximum a-posteriori probability over P.

Finally, let us define a functional $\mathcal{C}$, which extracts contextual information from P, producing a description of spatial distribution and relation between class probabilities: $\mathcal{C} : P \mapsto E \in \Re^{N_E}$.

The blocks $\mathcal{F}$, $\mathcal{P}$ and $\mathcal{C}$ can be combined to form the inference pipeline of an iterated Stacked Sequential Learning schema (see Fig.1). In Table 1, the training and inference algorithms of the Multi-class Iterated Multi-scale Stacked

TRAINING	INFERENCE

Require: $\mathbf{I}_{train}$, Y_{train}, N_{it}, $\mathcal{F}$, $\mathcal{C}$
Ensure: $\{\mathcal{P}^{(1)}, \mathcal{P}^{(2)}, \ldots, \mathcal{P}^{(\mathrm{it})}\}$
1: $F = \mathcal{F}(\mathbf{I}_{train})$
2: $E^{(1)} = \emptyset$;
3: **for** it $= 1{:}N_{\mathrm{it}}$ **do**
4:     $U^{(\mathrm{it})} = [F\ E^{(\mathrm{it})}]$
5:     $U_*^{(\mathrm{it})} = \text{subsampling}(U^{(\mathrm{it})})$
6:     LEARN $\mathcal{P}^{(\mathrm{it})}|_{\{U_*^{(\mathrm{it})}, Y\}}$
7:     **if** $it < N_{it}$ **then**
8:        $P^{(\mathrm{it})} = \mathcal{P}^{(\mathrm{it})}(U^{(\mathrm{it})})$
9:        $E^{(\mathrm{it}+1)} = \mathcal{C}(P^{(\mathrm{it})})$
10:    **end if**
11: **end for**

Require: $\{\mathcal{P}^{(1)}, \mathcal{P}^{(2)}, \ldots, \mathcal{P}^{(\mathrm{it})}\}$, I, N_{it}, $\mathcal{F}$, $\mathcal{C}$
Ensure: Y
1: $F = \mathcal{F}(I)$
2: $E^{(1)} = \emptyset$;
3: **for** it $= 1{:}N_{it}$ **do**
4:     $U^{(\mathrm{it})} = [F\ E^{(\mathrm{it})}]$
5:     $P^{(\mathrm{it})} = \mathcal{P}(U^{(\mathrm{it})})$
6:     **if** it $< N_{\mathrm{it}}$ **then**
7:        $E^{(\mathrm{it}+1)} = \mathcal{C}(P^{(\mathrm{it})})$
8:     **end if**
9: **end for**
10: $Y = \arg_c \max P^{(N_{\mathrm{it}})}$

Table 1: Training (left) and Inference (right) algorithm pseudocode.

Sequential Learning are detailed. It is worth to note that the main difference between the two stages is the learning of functional $\mathcal{P}$ (line 6, training).

The inference procedure consists of the following iterative classification procedure: given a CT scan slice, the functional $\mathcal{F}$ extracts appearance features (line 1); the functional $\mathcal{P}^{(1)}$ estimates the posterior probabilities $P^{(1)}$ (line 5); the functional $\mathcal{C}$ extracts the contextual information $E^{(2)}$ (line 7); this information is joined to the appearance features (line 4); the resulting data U is then fed to the next functional $\mathcal{P}^{(2)}$. As shown in Fig. 1, these step (except for the appearance feature extraction) can be iterated (gray square in the figure, lines from 3 to 9). The final step (line 10) outputs the final labeling Y as the pixelwise most probable class in $P^{(N_{\mathrm{it}})}$. An additional stage of *classes reduction* is depicted in Figure 1(f). This step is used just once in the framework, between the first and the second iteration, with the aim of fusing the *body* and *airspace* classes. This part is detailed next subsections, along with a detailed descriptions on the implementation of $\mathcal{F}$, $\mathcal{P}$ and $\mathcal{C}$.

Appearance features extraction, $\mathcal{F}$. The features of appearance used in this study are based on [14], where three Gaussian kernels have been defined, with standard deviations $(\lambda_1, \lambda_2, \lambda_3) = (0.43, 0.86, 1.3)\ mm$. The Gradient Magnitude and the Laplacian for every pixel in the slice are then computed at the three scales and used as features, along with the eigenvalues of the Hessian matrices. The pixel-wise HU value along with its smoothed value at the three scales is considered as well. A feature vector $F \in \Re^{16}$ is finally assigned to each pixel.

Probability estimation, $\mathcal{P}$. To build the functional $\mathcal{P}$, we deal with the multiclass problem using the Error Correcting Output Codes (ECOC) [15] and combining Support Vector Machine (SVM) with Intersection Kernel [16] as base classifier. The ECOC matrix is filled with {-1,1} values when class samples are

used as positive or negative samples, respectively, to train the corresponding classifier; zero value is used if no samples of that class are used to train the classifier. In order to classify a pixel with feature vector F, we test it with all the trained SVMs, and we join the results into a codeword. We compute then the distance d_{ECOC} between the codeword and each row of the ECOC matrix: a small distance indicates a high likelihood with respect to the corresponding class. The class pseudo-probability is estimated as $p(c) \propto \exp^{-\alpha d_{ECOC}(c)}$, where $\alpha = \ln(N_C)/\sqrt{N_C}$ is used, assuring a codeword equidistant from all the rows is also equiprobable with respect to all the classes.

Contextual feature extraction, $\mathcal{C}$. In the MS-SSL scheme, the functional $\mathcal{C}$ is implemented by using a multi-scale procedure: given a set of s spatial scales $\Sigma = \{\sigma_1, \sigma_2, \ldots, \sigma_s\}$, at each scale the label field probability P is filtered by a Gaussian isotropic spatial kernel with corresponding standard deviation, forming a set of filtered label field probabilities: $P_\sigma = P * G(0, \sigma)$. Given a spatial position (x_i, y_i), and for each $\sigma \in \Sigma$, P_σ is sampled on a regular grid based on a 8-neighborhood system plus the central pixel itself. The pixel displacement is proportional to the filtering scale σ; thus the displacement set, for a 2-D spatial support, can be defined as $\Delta = \sigma \cdot ([-1, 0, 1] \times [-1, 0, 1])$. This filtering and sampling scheme provides an interesting property: contextual information at long ranges is sampled from a properly smoothed version of the label probability field, so that the sampling is not noisy and is representative of a certain region, depending on the scale σ. Indeed, this sampling provides a good level of detail at small filtering scales, close to the central sampling point, and almost noiseless probability sampling at long range distances.

Gathering spatial contextual information in a reduced set of values, while including long range interaction is not a trivial task; a detailed discussion on the trade off between desired detail and long range coverage, with respect to the number of pixel-wise contextual features N_E can be found in [11]. The number of contextual features can be easily computed as the product of the displacement set cardinality by the number of spatial scales times the number of classes, i.e. $N_E = |\Delta||\Sigma|N_C$. In this paper, we define the spatial scales in octave, so that the set Σ can be defined as $\Sigma = \{1, 2, ..., 2^{i-1}, 2^{s-1}\}$. It is clear that the farthest sampling is 2^{s-1} pixels away from the central pixel. Depending on the image size S_x and S_y, a maximum value of s should be used, since sampling too far away (outside the image spatial support) becomes totally useless.

It is worth to note that the number of scales may vary along the iterations. In this way, the amount of contextual information at rough and fine image detail can be adapted to problem-dependent image domain and properties. In order to achieve a coarse-to-fine context description, reducing the risk of over fitting, we compute the context at all the scales just at the first iteration, e.g., $s|_{it=2} = 6$, while we consider only the smallest three scales in next iterations, e.g., $s|_{it>2} = 3$.

Classes reduction. As stated above, in our approach we define five regions in CT scans: *left lung, right lung, airways, body, air-space*, with labels $y =$

$(1, \ldots, 5)$, respectively. In order to properly encode contextual information, the method requires the definition of all the regions in the slice. From the point of view of the appearance, airspace and some lung regions may have similar local appearance. This may cause problems of over fitting during training, due to appearance features and when small scales are considered (e.g., some parts of airspace may be classified inside the body). From the point of view of context description, it is desirable that everything that is not lungs or airways is considered as a unique class during training. For this purpose, we implement the following additional step with the aim of merging the body and airspace classes: (1) the functional $\mathcal{P}^{(1)}$ is trained with five classes; (2) from the output $P^{(1)}$ we consider the region *body* as the set of pixels of the maximum a posteriori probability matrix with label $y = 4$; (3) we fill the *body* region with morphologic operations [17], and we consider the positions $\boldsymbol{q}_{ext} \notin body$ and $\boldsymbol{q}_{body} \in body$; (4) we compute the position $\boldsymbol{q}_{body}^{max}$ with maximum probability of class *body* in $P^{(1)}$; (5) we update the pixels $\{\boldsymbol{q}_{ext}\}$ with features similar to body, for the sake of subsequent training procedures: $P^{(1)}(y = 4)|_{\boldsymbol{q}_{ext}} = P^{(1)}(y = 4)|_{\boldsymbol{q}_{body}^{max}}$, and $F^{(1)}|_{\boldsymbol{q}_{ext}} = F^{(1)}|_{\boldsymbol{q}_{body}^{max}}$.

3   Experiments

Material. We use two different datasets, one for training and validation, and one testing. For training and validation purposes, we use a dataset that includes a large amount of challenging pathological cases acquired with scanners of four different vendors; from this dataset, we select two subsets of 250 CT slices (TRds), and 150 CT slices (VAds) from 40 patients. In each slice, the contour of *left lung*, *right lung*, *airways* and *body* were manually annotated (see Figure 2). The TRds dataset was used to train MiM-SSL using 10000 samples per class, randomly selected. For SVM inference we make use of the fast approximation provided in

Fig. 2: Annotation.

[16]. We avoid data intersection while training each stage of the iterations, to avoid system over-fitting on the training data. For testing purposes, the public LOLA dataset is used, consisting of 55 CT scans acquired from different patients with different scanners. In all the experiments, each image dimension was rescaled by a factor 2 (256x256 px), to speed up the computation.
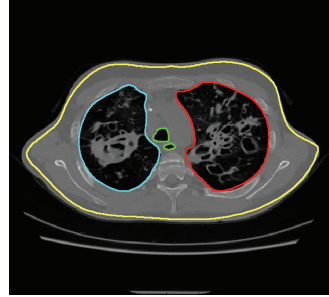
System setup. The contextual part of the proposed method has solely two parameters: the number of initial scales s and the number of iterations N_{it}. To optimize these two parameters we used the VAds dataset. The algorithm performance has been evaluated by means of the Jaccard measure, averaged between left and right lung, which is very sensitive to both false positive and false negative errors. The surface in Figure 3(left) shows the Jaccard measure as
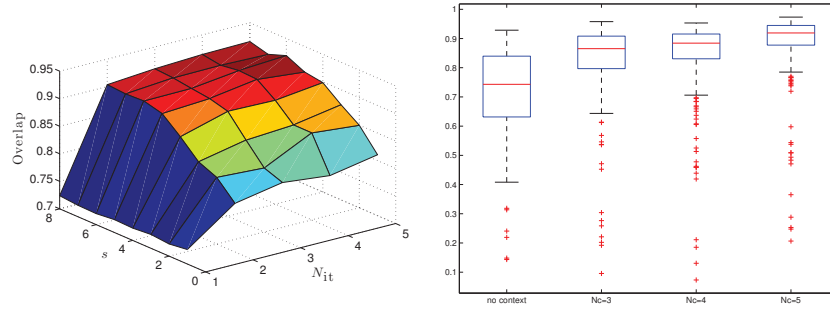
Fig. 3: Jaccard measure performance on the VAds varying N_{it} and s (left). Jaccard measure performance while adding semantic classes to the contextual features (right).

a function of the two parameters. The optimal parameters results to be $N_{it} = 5$ and $s = 6$, providing and average Jaccard measure of 91.91%. The surface clearly shows that increasing the number of scales s, which increases the maximum range of labels interaction, increases the algorithm performance. However, as expected, the performance with $s \geq 6$ reaches a plateau where is not clear anymore if increasing s, increases the performance; the contextual information available at $s = 6$ is sufficient to understand the implicit semantic class spatial relationship.

As it can be noticed, the contribution of context is significant; the appearance based classification ($N_{it} = 1$), provides a Jaccard of only 72.3%. With $s = 6$, the increment from $N_{it} = 2$ to $N_{it} = 5$ is of 1.65%; this is a substantial improvement since the Jaccard measure is very sensitive to errors, and the performance at $N_{it} = 2$ is already pretty high.

Finally, it is interesting to note that adding semantic classes makes the classification problem easier instead of more difficult, as also commented in [8]. Figure 3(right) shows the Jaccard performance when no context is used, when $N_c = 3$ classes (left and right lung, and rest) are used, when the airways are added ($N_c = 4$), and when the body class is also added ($N_c = 5$). The case $N_c = 5$ solely refers to the first iteration, since the technique of *classes reduction* is applied in this case. The progressive improvement is clearly visible; adding the body class substantially increases the system performance. A one-way ANOVA test at 5%, with multiple comparisons, ensures that the difference when adding the context ($N_c = 3$) w.r.t. no context is statistically significant (p-value $< 10^{-3}$), but also that adding the class body ($N_c = 5$) is statistically significant w.r.t. all previous configurations.

Test on LOLA dataset. Given the 55 scans of the LOLA 2011 challenge, we compute the features F for each pixel of each slice, and classify it independently from the rest of the scan. For this reason, no 3D information is exploited. The $\mathcal{P}$ functional is designed as a *one-vs-all* ECOC matrix, where SVMs are combined. Although the optimal number of iteration in system setup is $N_{it} = 5$, we experimentally observed comparable results with $N_{it} = 4$. For the sake of computation time, we performed our experiments with the latter configuration. As stated in

section 2, we use $s = 6$ for $it = 2$ (i.e., maximum distance interaction $\approx 10cm$), and then we decrease it to $s = 3$ for $it > 2$ (i.e., maximum distance interaction $\approx 1cm$). This has the effect of providing a fully contextual description at the initial stage, while concentrating on details in subsequent stages. Quantitative results for segmentation scores are shown in Table 2, compared with other approaches. For the sake of completeness, the method that won the challenge is also reported [18]. The proposed method (MiMSSL), achieves performance comparable with methods based on cognition-network [4], active-shape model [7], and the Keuhkot approach [5]. In most of the cases, the minimal value for right lung is 0.000, due to the scan 45, where the lung is not visible. The method in [8], which is based on a similar idea of MiMSSL, achieves better performance, but applies a final label refinement, which is not used in this paper.

Qualitative results on per-slice lung segmentation are depicted in Figure 4. We can observe that: (1) unlike some shape-based models, our approach is able to deal with a partially visible lung Fig. 4(m); (2) an accurate segmentation is achieved even in presence of lungs with fragmented appearance or non standard shape Fig. 4(g,i,l); (3) we achieve a good separation between lungs and external airways Fig. 4(a,f,g,h); (4) even without constrains on connectivity or specific rules, there is no confusion between air inside the lung and outside (e.g., trachea) Fig. 4(n,f); (5) the method automatically separates the two lungs: in case their separation is clearly visible, they are correctly separated, otherwise, the method tends to separate them in the middle of the image Fig. 4(b,c,f). This is due to the statistics of the position of the separation in the training set; (6) in some cases, regions that appear separated from the main lung region are also segmented as lung. Usually, these regions are located in the lower part of the abdomen. Although this phenomenon is a drawback in some cases Fig. 4(d), it may help in cases where the lung appears as separated into several regions, due to the horizontal cut of the scan Fig. 4(e). This issue could be solved by taking advantage of 3D information for classification in the future implementation of the method. A complete overview of the results on the whole dataset is depicted in Figure 5. In some cases (scan 7, 10, 27, 30, 34, 42, 51), areas outside the lungs are labelled as lung. This problem is due to the effect of contextual features at large scales, which in test set produce artifacts in areas where lung is expected to be. These features are also responsible for weak precision at lung borders in some cases.

	MiMSSL	[18]	[4]	[7]	[5]	[8]
score	0.949	0.973	0.949	0.949	0.948	0.952
mean	0.941/0.956	0.974/0.972	0.950/0.949	0.939/0.959	0.935/0.961	0.952/0.951
std	0.092/0.132	0.097/0.135	0.172/0.187	0.173/0.122	0.209/0.147	0.117/0.132
min	0.326/0.000	0.277/0.000	0.000/0.000	0.039/0.167	0.000/0.000	0.116/0.000
Q1	0.936/0.967	0.987/0.991	0.980/0.984	0.979/0.985	0.978/0.986	0.965/0.964
median	0.967/0.978	0.992/0.994	0.987/0.990	0.990/0.990	0.990/0.993	0.974/0.974
Q3	0.976/0.982	0.995/0.996	0.993/0.993	0.994/0.994	0.995/0.996	0.978/0.977
max	0.990/0.991	0.999/0.999	0.998/0.998	0.997/0.997	0.998/0.998	0.987/0.987

Table 2: Comparison on LOLA dataset results. The segmentation scores provided by the challenge organizers are reported.
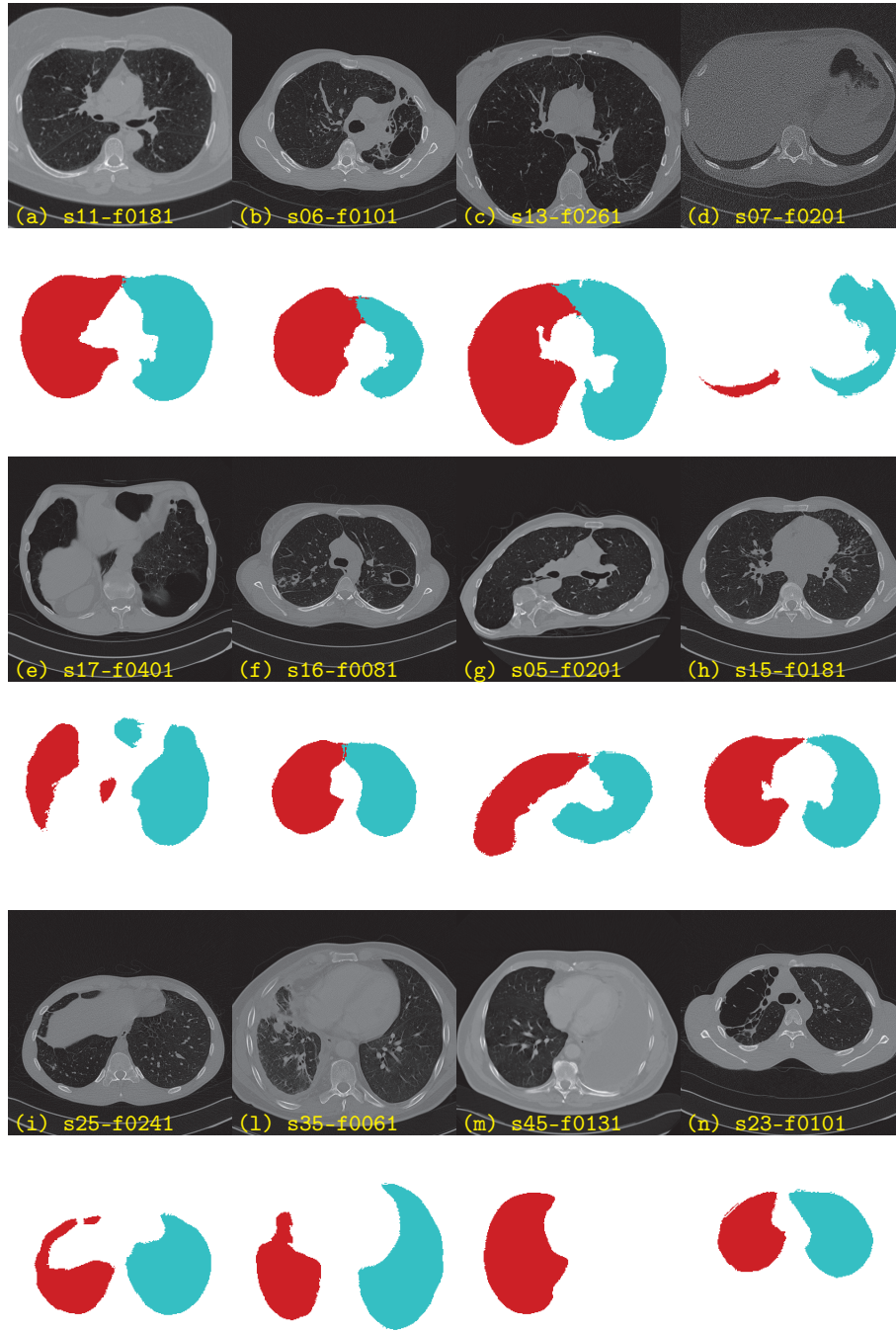
Fig. 4: Examples of automatic lung segmentation with MiMSSL. The CT scan is shown, along with the result, where in red and blue are represented the two lungs. The scan number and slice number in the LOLA2011 dataset is also indicated with the letters s and f, respectively.
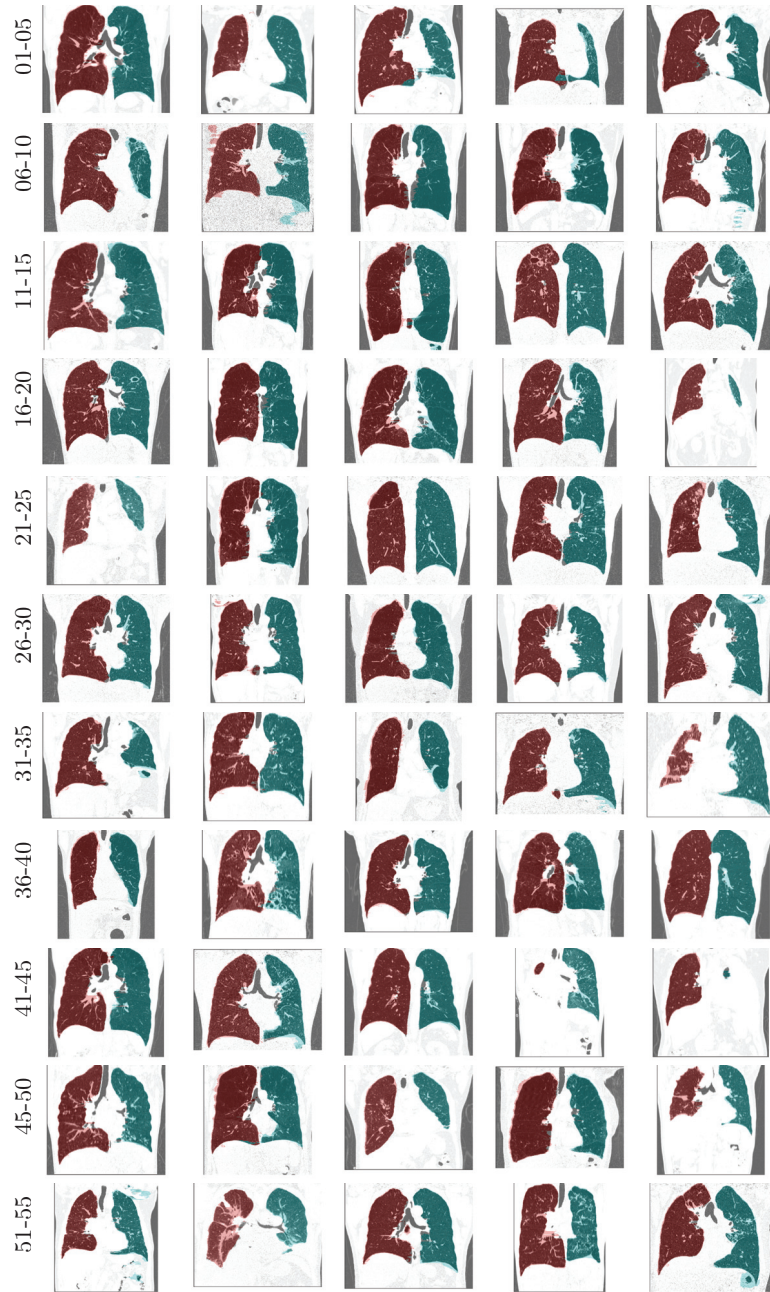
Fig. 5: Coronal slices of 55 LOLA2011 scans. The corresponding segmentation is also indicated, in red the left lung, in blue the right lung.

4   Discussion

The computational cost of the proposed algorithm depends on the number of image pixels N, the number of scales s, the number of classes N_c and the number of iterations N_{it}. Gaussian filtering is performed with the $\mathcal{O}(N)$ approximated method in [19]. Contextual information extraction ($\mathcal{C}$) is $\mathcal{O}(sNN_c(N_{it} - 1))$, requiring a Gaussian convolution per scale per class per iteration (except for the first one). Function $\mathcal{P}$ requires an SVM inference per class per iteration; since we use the approximate SVM inference in [16] which is $\mathcal{O}(N)$, computing $\mathcal{P}$ is $\mathcal{O}(NN_cN_{it})$. For these reasons, our proposal scales linearly on every parameter. Moreover, great part of the operations within one iterations can be parallelized, e.g. filtering and SVM inference among classes. A MATLAB non-parallelized version of the algorithm requires about 4.8 seconds per slice on a 2.8 GHz Intel Core i7 processor and 8GB of RAM; the averaged computation time for all the scans of LOLA dataset in a multi-core AMD Opteron 6272 at 2.1GHz with 64GB of RAM is of 5 minutes per scan.

   In terms of performance (see Figure 3), it has to be noted that without the contextual features ($N_{it} = 1$), the system perform poorly since there is no clue to discriminate between left and right lung. Adding a stage that uses contextual information ($N_{it} = 2$) increases the performance in a significant way only if the number of scales s is greater than 4. This is due to the fact that increasing the number of scales allows to model long-range interaction between classes. Increasing the number of iterations, increases the Jaccard measure; however, there is a limitation in the improvement and thus further stages do not provide significant improvement while increasing the computational time.

   Finally, it is worth to note that in this work we used basic descriptors as features. Since the method is based on classification, we expect using richer texture descriptors (e.g. LBP) further improves the overall performance. This task is out of scope for the current paper, where our aim is to demonstrate the usefulness of the presented general approach in this specific segmentation problem.

5   Conclusions and future work

We have presented a fully automatic method for lung segmentation in Computed Tomography, which implicitly embeds contextual information and efficiently encodes long term interactions between pixels. The method has been applied to the LOLA2011 lung segmentation dataset. A straightforward improvement in the method is the use of 3D information along the whole scan, which could solve the issue (6) in section 3. However, this requires a sufficient number of 3D ground truth volumes. Furthermore, an analysis of feature importance in the sparse matrix of contextual information is necessary, especially in 3D extension. The same method can be applied to other segmentation problems in both CT and MRI, such as liver, etc. Future works encompass the refinement of lung segmentation; as an example, in cases where the lungs are in contact, the separation between left and right lung is visible as a thin line of tissue and it is very difficult to separate the lungs at the appropriate position by means of a learning approach. In this cases, local refinement seems to be mandatory.

References

 1. Rikxoort, E.M., de Hoop, B., Viergever, M.A., Prokop, M., van Ginneken, B.: Automatic lung segmentation from thoracic computed tomography scans using a hybrid approach with error detection. Medical Physics **36(7)** (2009) 2934–2947
 2. Brown, M., McNitt-Gray, M., Mankovich, N., Goldin, J., Hiller, J., Wilson, L., Aberle., D.: Method for segmenting chest ct image data using an anatomical model: preliminary results. IEEE Trans Med Imaging **16(6)** (1997) 828–839
 3. Weinheimer, O., Achenbach, T., Heussel, C.P., Duber, C.: Automatic lung segmentation in mdct images. Technical report, LOLA Challenge 2011 (2011)
 4. Korn, R., Kim, J., Schmidt, G., Binnig, G.: Description of a fully automatic lung segmentation algorithm based on the cognition network technology. Technical report, LOLA Challenge 2011 (2011)
 5. Pinho, R., Delmon, V., Vandemeulebroucke, J., Rit, S., Sarrut, D.: Keuhkot: A method for lung segmentation. Technical report, LOLA Challenge 2011 (2011)
 6. Kuhnigk, J.M., Dicken, V., andL. Bornemann, S.Z., Kuemmerlen, B., Krass, S., Peitgen, H.O., Yuval, S., Jend, H.H., Rau, W.S., Achenbach, T.: New tools for computer assistance in thoracicct - part i: Functional analysis of lungs, lung lobes, and bronchopulmonary segments. RadioGraphics **25(2)** (2005) 525–536
 7. Sun, S., Bauer, C., Beichel, R.: Robust active shape model based lung segmentation in ct scans. In: LOLA Challenge 2011. (2011)
 8. Montillo, A.: Context selective decision forests and their application to lung segmentation in ct images. In: Workshop in Pulmonary Image Analysis. (2011)
 9. Loog, M., van Ginneken, B.: Supervised segmentation by iterated contextual pixel classification. In: In Proceedings 16th International Conference on Pattern Recognition. (2002) 925–928
10. Cohen, W.W.: Stacked sequential learning. In: International Joint Conference on Artificial Intelligence. (2005) 671–676
11. Gatta, C., Puertas, E., Pujol, O.: Multi-scale stacked sequential learning. Pattern Recognition **44**(10-11) (2011) 2414–2426
12. Seyedhosseini, M., Kumar, R., Jurrus, E., Giuly, R., Ellisman, M.H., Pfister, H., Tasdizen, T.: Detection of neuron membranes in electron microscopy images using multi-scale context and radon-like features. In: MICCAI (1). (2011) 670–677
13. Ciompi, F., Pujol, O., Gatta, C., Alberti, M., Balocco, S., Carrillo, X., Mauri-Ferre, J., Radeva, P.: Holimab: A holistic approach for media-adventitia border detection in intravascular ultrasound. Medical Image Analysis **16**(6) (2012) 1085–1100
14. Sluimer, I.C., Prokop, M., van Ginneken, I.H.B.: Automated classification of hyperlucency, fibrosis, ground glass, solid, and focal lesions in high-resolution ct of the lung. Medical Physics **33(7)** (2006) 2610–2620
15. Dietterich, T.G., Bakiri, G.: Solving Multiclass Learning Problems via Error-Correcting Output Codes. JAIR **2** (1995) 263–286
16. Maji, S., Berg, A.C., Malik, J.: Classification using intersection kernel support vector machines is efficient. In: CVPR. (2008) 1–8
17. Soille, P.: Morphological image analysis: Principles and applications. Springer-Verlag (1999) 173–174
18. Lassen, B., Kuhnigk, J.M., Schmidt, M., Krass, S., Peitgen, H.O.: Lung and lung lobe segmentation methods at fraunhofer mevis. In: LOLA 2011 Challenge. (2011)
19. Geusebroek, J.M., Smeulders, A.W.M., van de Weijer, J.: Fast anisotropic gauss filtering. IEEE Transactions on Image Processing **12**(8) (2003) 938–943