Deep architectures for high-resolution multi-organ chest X-ray image segmentation

Oscar Gómez, Pablo Mesejo, Oscar Ibáñez, Andrea Valsecchi, Oscar Cordón

Noname manuscript No.
(will be inserted by the editor)

Deep architectures for high-resolution multi-organ chest X-ray image segmentation

**Oscar Gómez · Pablo Mesejo ·
Oscar Ibáñez · Andrea Valsecchi ·
Oscar Cordón**

the date of receipt and acceptance should be inserted later

Abstract Chest X-ray images (CXRs) are the most common radiological examination tool for screening and diagnosis of cardiac and pulmonary diseases. The automatic segmentation of anatomical structures in CXRs is critical for many clinical applications. However, existing deep models work on severely down-sampled images (commonly 256 × 256 pixels), reducing the quality of the contours of the resulting segmentation and negatively affecting the possibilities of such methods to be effectively used in a real environment. In this paper, we study multi-organ (clavicles, lungs, and hearts) segmentation, one of the most important problems in semantic understanding of CXRs. We completely avoid down-sampling in images up to 1024 × 1024 (as in the JSRT dataset) and we diminish its impact in higher resolutions via network architecture simplification without a significant loss in the accuracy. To do so, we propose four different convolutional models by introducing structural changes to the baselines employed (U-Net and InvertedNet) as well as by integrating several techniques barely used by CXRs segmentation algorithms, such as instance normalization and atrous convolution. We also compare single-class and multi-class strategies to elucidate which approach is the most convenient for this problem. Our best proposal, X-Net+, outperforms 9 state-of-the-art methods on clavicles and lungs obtaining a Dice similarity coefficient of 0.938 and 0.978, respectively, employing a 10-fold cross validation protocol. The same architecture yields comparable results to the state-of-the-art in heart segmentation with a Dice value of 0.938. Finally, its reduced version, RX-Net+, obtains similar results but with a significant reduction in memory usage and training time.

Keywords Semantic segmentation · chest x-ray segmentation · convolutional neural networks · deep networks simplification

All the authors are with the Andalusian Research Institute in Data Science and Computational Intelligence (DaSCI), University of Granada, Granada, Spain. ·
All the authors are with the Department of Computer Science and Artificial Intelligence, University of Granada, Granada, Spain. ·
Pablo Mesejo is also with the Perception team at Inria Grenoble Rhône-Alpes, Grenoble, France. ·
Oscar Ibáñez is also with Panacea Cooperative Research, Ponferrada, Spain. ·
Corresponding author email: ogomez@decsai.ugr.es (O. Gómez)

1 Introduction

X-ray images represent the most commonly employed medical imaging modality [1]. In particular, chest X-rays (CXRs) are the most commonly performed radiology examination world-wide [2] because they are able to produce images of the heart, lungs, airways, blood, vessels, spine, and chest [3], and because of their diagnosis and treatment potential [2, 4]. In order to quantify the importance of CXR analysis, it is important to notice that 2.02 million CXRs were performed in 2015/16 by the National Health Service of United Kingdom [5], and that 150 million CXRs are annually acquired in the United States alone[6].

Among all anatomical structures displayed in CXRs, the lungs, the heart and the clavicles are particularly important. Lungs radiography has been largely used for the diagnosis of pulmonary diseases such as pneumonia, tuberculosis, emphysema, and lung cancer [3, 7, 8]. Hearts are often employed to detect cardiovascular disease such as chronic stable angina or valvular heart disease [9]. Lastly, clavicles are used to detect lesions, such as tumorous lesions [10], or for forensic identification [11] via comparing their silhouette in ante-mortem and post-mortem X-ray images. Despite the great clinical importance of X-ray image understanding, a task such as the segmentation of anatomical structures in CXRs remains very challenging. This is mainly due to the projective nature of X-ray imaging, which causes large overlapping of anatomies, fuzzy object boundaries and complex texture patterns. For instance, even among expert radiologists, minor errors in diagnosis are performed in *circa* 30% of studies [12] and major errors in 3-6% [13, 14]. CXRs are a cornerstone of acute triage as well as longitudinal surveillance. Despite the ubiquity of the exam and its apparent technical simplicity, CXRs are widely regarded among radiologists as among the most difficult to master [13].

The automatic CXRs segmentation has been extensively studied since the ’70s, at least regarding the segmentation of lungs, rib cage, heart, and clavicles [15, 16]. Conventional methods rely on prior knowledge [17] to delineate anatomical objects from X-ray images. Modern approaches utilize deep convolutional networks and have shown superior performance [18]. More than 150 research works dealing with this problem were already published during the twentieth century [19], raising the number to 331 at present, according to Scopus[1]. Most works have focused on the segmentation of a single organ, mainly the lungs [8, 20, 21] for its medical importance[2]; followed by the heart [22], where most works plainly extrapolate the approaches used for lungs; and lastly the clavicles [23], the organ whose segmentation entails greater difficulty (reflected in a lower quality of the final segmentation [24]). Despite the great advances made in the automatic segmentation of these organs, limitations still persist, such as the need to use down-sampled CXRs or the irregularity and imprecision of the edges resulting from segmentation, which reduce their applicability in clinical settings.

In this paper, we tackle the segmentation of hearts, clavicles and lungs in CXRs using convolutional neural networks (ConvNets). For this study, we have used a public database of CXRs called JSRT [25] with its associated ground truth [26].

[1] Search performed the 8th September 2018 using the keywords (TITLE-ABS-KEY (chest AND x-ray AND segmentation) OR TITLE-ABS-KEY (chest AND radiograph AND segmentation) AND NOT TITLE-ABS-KEY (computed AND tomography))

[2] According to the International Agency for Research on Cancer, lung cancer was the most common cause of cancer death in 2015 with 1.69 million deaths.

The main goal of this paper is threefold. First, to improve the state-of-the-art results in CXR segmentation. To do so, we introduce a new architecture, called X-Net, that incorporates structural changes in the network architecture as well as integrates several techniques barely used by CXRs segmentation algorithms, such as instance normalization [27] and atrous convolution (a.k.a. dilated convolution) [28]. These modifications allow us to improve the segmentation accuracy of the state of the art. Further structural modifications (resulting in an extension termed X-Net+) also allow us to work with images up to 1024×1024 in only one GPU (an example of a segmentation obtained by X-Net+, our best network, is shown in the Figure 1). Second, to propose a simplification of X-Net and X-Net+, called RX-Net (Reduced X-Net) and RX-Net+, respectively, that reduce even more both memory usage and training time, while keeping similar results. Third, to investigate single-class (a ConvNet is trained to segment each organ separately) and multi-class (a ConvNet is trained to segment all organs simultaneously) segmentation approaches to elucidate which one is more suitable for the task at hand.

This paper is structured as follows. Section 2 reviews the current state of the art in CXRs segmentation. Section 3 describes our proposals. Section 4 presents experiments and results. The conclusions are detailed in Section 5.

2 Related works

Many different taxonomies can be presented to classify image segmentation approaches [19, 29, 30, 31]. One possible classification could be the following: (1) rule-based methods, where the image is segmented by the application of a set of low-level and spatially blind rules (such as thresholding, edge detection, or region growing) [19]; (2) shape-based methods [32], where the segmentation is performed by matching a model, that includes some sort of prior shape information, to the image (such as active shape models, or active appearance models); (3) atlas-based methods [33], generally based on the registration of an atlas (i.e. an already segmented image) and a target image; (4) graph-based methods [34], that represent the image as a graph that is partitioned into a set of separated connected components (generally making use of techniques such as conditional random fields or Markov random fields); (5) machine learning-based methods [31], traditionally based on handcrafted features (e.g. SIFT) together with a classifier (i.e. k-NN or an artificial neural network) but, with the advent of ConvNets [35], this paradigm has shifted towards end-to-end approaches where the ConvNet input is directly the image to segment and the output is the target segmentation. Since each methodology has its own pros and cons, the best results are commonly achieved via hybrid approaches that combine two or more of the previous strategies [36, 37]. In general terms, in the case of CXRs segmentation, most approaches are either rule-based [19], shape-based [26], or machine learning-based [8]. Given that the state of the art in CXRs segmentation are deep learning techniques, and ConvNets in particular, this paper delves into this research line.

The Japanese Society of Radiological Technology (JSRT) [25], in cooperation with the Japanese Radiological Society (JRS), created the standard digital image database with and without chest lung nodules (JSRT dataset) in 1998. Since then, the JSRT dataset has been used by a number of researchers in the world for various research purposes such as image processing, image compression, and computer-

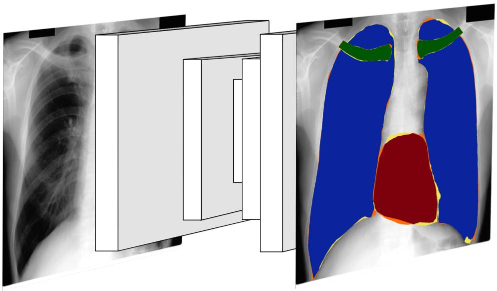

Fig. 1 Example of a segmentation obtained by X-Net+ with the overlap between the ground-truth and the segmentation in green, blue, and red for the clavicle, lungs, and heart, respectively; the over-segmented area in orange, and the under-segmented one in yellow. This segmentation obtains errors of 0.964, 0.931 and 10 for the DSC, JI and HD measures (detailed in the Section 4.2), respectively, with the clavicle, 0.965 and 16.763 with heart, and 0.985 and 53.225 with the lungs.

aided diagnosis. In particular, the JSRT dataset represents the most popular dataset for CXRs segmentation, including high resolution images (2048×2048 size, 0.175mm pixel size) and high resolution segmentation masks (provided by [26]). Those segmentation masks have a resolution of 1024×1024 (i.e. ground truth resolution). Methods tested on JSRT are commonly evaluated using images of smaller resolution (256×256).

The state of the art in lungs segmentation with the JSRT dataset [36] is based on a hybrid approach with four stages devoted to: 1) preprocessing the X-ray images by augmenting the contrast between the lungs and their surrounding area; 2) extracting the foreground (which incorporates the upper torso region) by using an intelligent block-based binarization; 3) excluding lung regions from the foreground through a series of spatial-based processing operations; and 4) employing an adaptive graph cut technique to locally refine the preliminary lung boundaries. On the other hand, the state of the art in hearts [38] and clavicles [24] segmentation is based on deep learning approaches that we describe below.

With the advent of deep learning, many works have employed deep models to achieve state-of-the-art results in many problems, such as gait-pattern classification [39], object tracking [40], image classification [41], electroencephalogram classification [42], and semantic segmentation [35, 43], just to name few examples. In particular, CXRs segmentation methods based on ConvNets have outperformed prior art based on classical techniques (corresponding to the first four categories in the aforementioned taxonomy). In particular, CXRs segmentation methods based on ConvNets have outperformed prior art based on classical techniques (corresponding to the first four categories in the aforementioned taxonomy). Firstly, an encoder-decoder network called U-Net [18] was studied for multi-class segmentation of lungs, heart and clavicles achieving comparable, or higher, accuracy on most of the structures when compared with the state-of-the-art segmentation methods [44]. They also studied the differences between multi-class and single-class training approaches, showing that for U-Net a multi-class approach helps the deep neural network to converge faster and deliver better segmentation results on the clavicles than the single-class. However, network outputs present holes inside the targeted structures as well as artifacts (i.e. small isolated segmented areas), which were solved with a post-processing step based on a level-set method. Afterwards, another work proposed a small modification of U-Net, called LF-SegNet [45]. LF-SegNet modified the up-sampling strategy, incorporated normalization techniques such as batch normalization [46], and employed data augmentation, slightly improving the performance on lungs segmentation in both the JSRT dataset and the Montgomery dataset [47] (that includes 138 CXRs and ground truth only for the lungs). Very recently, several articles tackled the segmentation of CXRs employing fully ConvNets [48, 44]. Also, a generative adversarial network approach called dual-path adversarial learning (DAL) based on a hybridization of a fully convolutional network and U-Net [38] was proposed for different kinds of medical image segmentation problems. DAL was tested for the segmentation of lungs and hearts, trained with images of 512 × 512 and evaluated on images on the ground truth resolution 1024 × 1024 resulting in the state of the art for heart segmentation.

A work closely related to ours was published by Novikov et al. [24] in 2018, where a modification of U-Net, called InvertedNet (INET), segmented the three organs and achieved state-of-the-art results for clavicle segmentation. INET outperformed prior art by reducing the number of filters per convolutional layer, therefore decreasing the possibility of over-fitting. Furthermore, motivated by the Gaussian noise inherited from the X-ray images acquisition process, INET added Gaussian dropout layers [49] and utilized a weighted loss function based on the Dice Similarity Coefficient (DSC) [50]. Gaussian dropout is a generalization of dropout where the activations are multiplied with random variables drawn from other distributions. This new form of dropout amounts to adding a Gaussian distributed random variable with zero mean and standard deviation equal to the activation of the unit. INET only considers the multi-class approach and do not compare with training the network for only one class. Finally, it can only be used with down-sampled images (with a resolution of 256 × 256 pixels) and the main option pointed by the authors to operate with higher resolution images was the use of a multi-GPU scenario, unlike in this work where we opted for the modification of the network architecture.

To the best of our knowledge, there are no other works in the literature that apply the atrous convolution [28] to the segmentation of CXRs, while it is one of the key elements of the state-of-the-art network for image segmentation in general [43]. Atrous convolution is a convolutional operation that introduces a spacing between the values in a kernel (the number of spaces between values is called dilated rate). This allows to adjust filter's field-of-view and capture multi-scale context information without reducing the spatial dimensions of the features maps (i.e. a 3 × 3 kernel with a dilation rate of 2 will have the same field-of-view as a 5 × 5 kernel, while only using 9 parameters). However, this is computationally expensive and takes a lot of memory, as a consequence its use is normally preceded by a few pooling layers to make the features maps computationally approachable as in DeepLab [43]. Also, we are not aware of other works studying the compression/simplification of deep networks, devoted to medical image segmentation, for their deployment in single-GPU devices or to allow them to work with larger images. Many researchers have pointed out that ConvNets suffer from heavy over-parameterization and can be efficiently reconstructed with only a small subset of its original parameters [51]. Therefore, several works have been published studying the simplification/compression of ConvNets reducing the required resources without significant loss in the original accuracy [52, 53]. Furthermore, we also perform a comparison between multi-class and single-class approaches using a k-fold cross-validation protocol, which is a much more rigorous evaluation strategy than the one commonly employed in the deep learning literature (where generally a simple hold-out is used).

3 Methodology

3.1 Architectures

The deep architectures proposed in this paper are inspired by INET [24] (described in Section 2 and depicted in Fig. 2) which, in turn, is a modification of U-Net. INET is devoted to the segmentation of lungs, hearts and clavicles in CXRs, and represents the state of the art in clavicle segmentation.

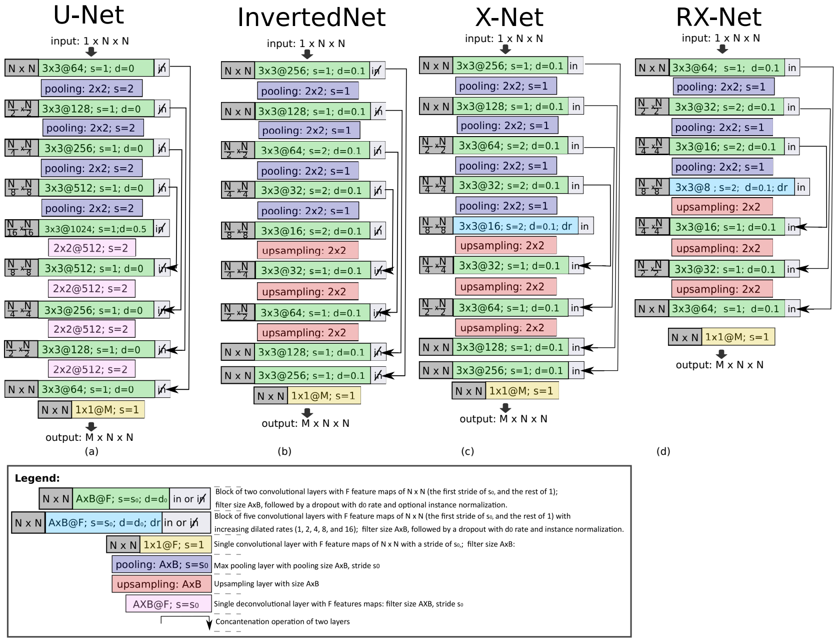
U-Net
input: 1 x N x N
N x N 3x3@64; s=1; d=0
pooling: 2x2; s=2
N/2 x N/2 3x3@128; s=1; d=0
pooling: 2x2; s=2
N/4 x N/4 3x3@256; s=1; d=0
pooling: 2x2; s=2
N/8 x N/8 3x3@512; s=1; d=0
pooling: 2x2; s=2
N/16 x N/16 3x3@1024; s=1;d=0.5
2x2@512; s=2
N/8 x N/8 3x3@512; s=1; d=0
2x2@512; s=2
N/4 x N/4 3x3@256; s=1; d=0
2x2@512; s=2
N/2 x N/2 3x3@128; s=1; d=0
2x2@512; s=2
N x N 3x3@64; s=1; d=0
N x N 1x1@M; s=1
output: M x N x N
(a)
InvertedNet
input: 1 x N x N
N x N 3x3@256; s=1; d=0.1
pooling: 2x2; s=1
N x N 3x3@128; s=1; d=0.1
pooling: 2x2; s=1
N/2 x N/2 3x3@64; s=2; d=0.1
pooling: 2x2; s=1
N/4 x N/4 3x3@32; s=2; d=0.1
pooling: 2x2; s=1
N/8 x N/8 3x3@16; s=2; d=0.1
upsampling: 2x2
N/4 x N/4 3x3@32; s=1; d=0.1
upsampling: 2x2
N/2 x N/2 3x3@64; s=1; d=0.1
upsampling: 2x2
N x N 3x3@128; s=1; d=0.1
N x N 3x3@256; s=1; d=0.1
N x N 1x1@M; s=1
output: M x N x N
(b)
X-Net
input: 1 x N x N
N x N 3x3@256; s=1; d=0.1 in
pooling: 2x2; s=1
N x N 3x3@128; s=1; d=0.1 in
pooling: 2x2; s=1
N/2 x N/2 3x3@64; s=2; d=0.1 in
pooling: 2x2; s=1
N/4 x N/4 3x3@32; s=2; d=0.1 in
pooling: 2x2; s=1
N/8 x N/8 3x3@16; s=2; d=0.1; dr in
upsampling: 2x2
N/4 x N/4 3x3@32; s=1; d=0.1 in
upsampling: 2x2
N/2 x N/2 3x3@64; s=1; d=0.1 in
upsampling: 2x2
N x N 3x3@128; s=1; d=0.1 in
N x N 3x3@256; s=1; d=0.1 in
N x N 1x1@M; s=1
output: M x N x N
(c)
RX-Net
input: 1 x N x N
N x N 3x3@64; s=1; d=0.1 in
pooling: 2x2; s=1
N/2 x N/2 3x3@32; s=2; d=0.1 in
pooling: 2x2; s=1
N/4 x N/4 3x3@16; s=2; d=0.1 in
pooling: 2x2; s=1
N/8 x N/8 3x3@8 ; s=2; d=0.1; dr in
upsampling: 2x2
N/4 x N/4 3x3@16; s=1; d=0.1 in
upsampling: 2x2
N/2 x N/2 3x3@32; s=1; d=0.1 in
upsampling: 2x2
N x N 3x3@64; s=1; d=0.1 in
N x N 1x1@M; s=1
output: M x N x N
(d)
Legend:
N x N AxB@F; s=s0; d=d0 in or in̸
Block of two convolutional layers with F feature maps of N x N (the first stride of s0, and the rest of 1); filter size AxB, followed by a dropout with d0 rate and optional instance normalization.
N x N AxB@F; s=s0; d=d0; dr in or in̸
Block of five convolutional layers with F feature maps of N x N (the first stride of s0, and the rest of 1) with increasing dilated rates (1, 2, 4, 8, and 16); filter size AxB, followed by a dropout with d0 rate and instance normalization.
N x N 1x1@F; s=1
Single convolutional layer with F feature maps of N x N with a stride of s0,; filter size AxB:
pooling: AxB; s=s0
Max pooling layer with pooling size AxB, stride s0
upsampling: AxB
Upsampling layer with size AxB
AXB@F; s=s0
Single deconvolutional layer with F features maps: filter size AXB, stride s0
Concantenation operation of two layers

Fig. 2 Schematic view comparing two preceding deep networks (a and b), and two of the deep networks proposed in this paper (c and d). All architectures employ Gaussian dropout, except U-Net that uses conventional dropout. 'in' stands for 'instance normalization'.

Even if INET currently offers the best performance in CXRs multi-class segmentation, it does not include some very relevant advances made in the image segmentation research field (atrous convolution) and deep learning in general (instance normalization). Consequently, the first methodological contribution of this paper is the introduction of a new architecture, called X-Net, that aims to increase the accuracy of INET through the inclusion of these advances. X-Net takes its name both from being designed to segment X-ray images and from the shape of the network (where, as usual, there is an encoding stage, which provides a reduced dimensionality representation of the input, and a decoding stage, that allows to recover an output of equal size to the input). First, X-Net takes advantage of instance normalization [27] at the end of each convolutional layer to add a normalization factor, accelerate the training, improve generalization, and reduce the dependency on the weights initialization. Second, X-Net changes the two central convolutional layers of INET by five atrous convolutional layers with increasing dilated rates of 1, 2, 4, 8 and 16 (see Fig. 2). These specific dilated rates are the most commonly employed in recent literature [54, 55, 56]. Atrous convolutions are convolutions with upsampled filters, that allow to enlarge the field-of-view and, therefore, to take into account more contextual information. The combined use of atrous convolution and instance normalization leads to a greater gain in performance than when they are employed separately (detailed in Section 4.4).

The second proposal introduced in this paper, termed Reduced X-Net (RX-Net), consists of a simplification of X-Net with the aim of obtaining similar accuracy but with a significantly lower memory usage and training time. Importantly, the main source of memory usage is not the trainable parameters, but instead the feature maps. Reducing the number of features maps of resolution $N \times N$ will result in a large memory reduction. RX-Net represents one alternative that makes possible experimenting with images of higher resolution than the one generally used, as we will discuss in the next paragraph. Therefore, the simplification involves the elimination of the first and last convolutional blocks of X-Net (notice that the first and last layers have the largest activation maps), and the reduction to half the number of convolutional filters of each convolutional layer (see Fig. 2), since these changes lead to the larger reduction in the ConvNet memory usage with the smaller drop in accuracy.

One important goal in CXRs image segmentation is to design a network able to work without any down-sampling or at least to reduce it to the minimum possible. The objective is to avoid upsampling the results (or diminish its impact) since upsampling causes a loss of detail in the final segmentation. In this sense, our next proposals are able to manage images of up to 1024×1024 in a single GPU (see subsection 4.3 for the technical details) without any down-sampling. Thus, these proposals are able to work with the ground truth resolution [26] of JSRT [25]. INET and X-Net would require too much memory or a multi-GPU scenario, being only able to work with images up to 256×256 (that is the resolution in which INET results, and most of the results in literature, are reported [24]). Our RX-Net can handle images up to 1024×1024, four times the usual resolution, but changing the input resolution results in a change in the relation between filter's field-of-view and feature maps, which would lead to a different behaviour than the one showed by RX-Net with images of 256×256. To avoid this drawback, a new architecture called RX-Net+ is proposed. This network is an incremental step from RX-Net maintaining all their layers (except the last convolutional block) with the same resolution (i.e. the value $N/4$ of RX-Net+ is equal to 256).

It allows us to employ the weights resulting from training RX-Net with 256×256 in RX-Net+ for images of 1024×1024, which significantly reduces the total training time. Additionally, RX-Net+ adds two pooling layers at the beginning (they could be replaced by convolutional layers but, if the number of feature maps introduced as input to the pre-trained RX-Net block is different to one, it would make impossible to re-use the weights from RX-Net in RX-Net+), and two final convolutional blocks connected to the inputs of the first two pooling layers (see Fig.

2). Furthermore, performing the pooling within the ConvNet allows to pass high-resolution information to the final layers of the ConvNet (see Fig. 2). Notice that comparing RX-Net and RX-Net+ for images of 1024 × 1024 will allow to study the importance of the relation between the filter's field-of-view and the feature maps. This approach could also be applied to X-Net giving rise to X-Net+. X-Net+ combines all the advantages of X-Net as well as allows to work with images of 1024 × 1024 in just one GPU. Training time and accuracy for both X-Net+ and RX-Net+ are improved thanks to training X-Net and RX-Net, respectively, then re-using the central block of common weights, and finally employing a simple fine-tuning. We will refer to our proposed deep networks as X-Net architectures, since they all are based on X-Net.

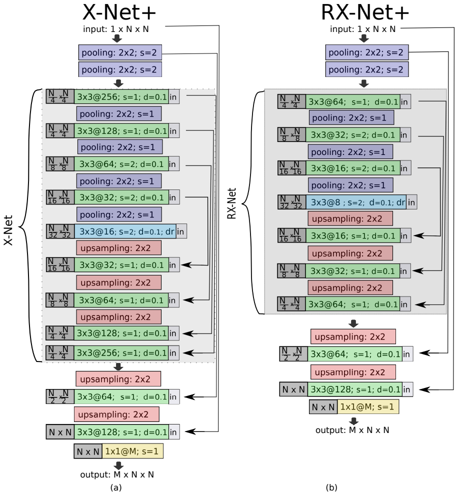
X-Net+
input: 1 x N x N
pooling: 2x2; s=2
pooling: 2x2; s=2
X-Net
$\frac{N}{4}$x$\frac{N}{4}$ 3x3@256; s=1; d=0.1 in
pooling: 2x2; s=1
$\frac{N}{4}$x$\frac{N}{4}$ 3x3@128; s=1; d=0.1 in
pooling: 2x2; s=1
$\frac{N}{8}$x$\frac{N}{8}$ 3x3@64; s=2; d=0.1 in
pooling: 2x2; s=1
$\frac{N}{16}$x$\frac{N}{16}$ 3x3@32; s=2; d=0.1 in
pooling: 2x2; s=1
$\frac{N}{32}$x$\frac{N}{32}$ 3x3@16; s=2; d=0.1; dr in
upsampling: 2x2
$\frac{N}{16}$x$\frac{N}{16}$ 3x3@32; s=1; d=0.1 in
upsampling: 2x2
$\frac{N}{8}$x$\frac{N}{8}$ 3x3@64; s=1; d=0.1 in
upsampling: 2x2
$\frac{N}{4}$x$\frac{N}{4}$ 3x3@128; s=1; d=0.1 in
$\frac{N}{4}$x$\frac{N}{4}$ 3x3@256; s=1; d=0.1 in
upsampling: 2x2
$\frac{N}{2}$x$\frac{N}{2}$ 3x3@64; s=1; d=0.1 in
upsampling: 2x2
N x N 3x3@128; s=1; d=0.1 in
N x N 1x1@M; s=1
output: M x N x N
(a)
RX-Net+
input: 1 x N x N
pooling: 2x2; s=2
pooling: 2x2; s=2
RX-Net
$\frac{N}{4}$x$\frac{N}{4}$ 3x3@64; s=1; d=0.1 in
pooling: 2x2; s=1
$\frac{N}{8}$x$\frac{N}{8}$ 3x3@32; s=2; d=0.1 in
pooling: 2x2; s=1
$\frac{N}{16}$x$\frac{N}{16}$ 3x3@16; s=2; d=0.1 in
pooling: 2x2; s=1
$\frac{N}{32}$x$\frac{N}{32}$ 3x3@8 ; s=2; d=0.1; dr in
upsampling: 2x2
$\frac{N}{16}$x$\frac{N}{16}$ 3x3@16; s=1; d=0.1 in
upsampling: 2x2
$\frac{N}{8}$x$\frac{N}{8}$ 3x3@32; s=1; d=0.1 in
upsampling: 2x2
$\frac{N}{4}$x$\frac{N}{4}$ 3x3@64; s=1; d=0.1 in
upsampling: 2x2
$\frac{N}{2}$x$\frac{N}{2}$ 3x3@64; s=1; d=0.1 in
upsampling: 2x2
N x N 3x3@128; s=1; d=0.1 in
N x N 1x1@M; s=1
output: M x N x N
(b)

Fig. 3 Schematic view of two of the deep networks proposed in this paper. These networks are extension of X-Net and RX-Net, respectively, allowing to handle ground truth resolution images (i.e. 1024 × 1024) in just one GPU. The legend of this figure can be seen in Fig. 2.

3.2 Training strategies

Two strategies are compared to train all networks: a single-class approach (to train a network to segment only one organ, i.e. three different networks are required to segment the three organs), and a multi-class approach (to train a network to jointly segment the three organs). The loss function in the single-class approach is directly the usual DSC [50] (see Section 4.2), and for the multi-class segmentation we employ a balanced version of this measure, defined as the product of the DSC values obtained for each single organ, i.e. DSC $= \prod_{i=1}^{n} DSC_i$, where n is the number of classes to segment, and DSC_i is the DSC value for the segmentation of the class i (corresponding to each organ to segment). This loss function allows to deal with the imbalanced nature of the CXRs segmentation problem (for instance, in the JSRT dataset ground truth[26], the 73.53%, 21.85%, and 4.62% of image pixels on average belong to lungs, hearts and clavicles, respectively) looking for solutions that properly segment the three classes. This loss function is stricter than others employed in the state of the art (for instance, the weighted mean in [24]) because we intend to encourage solutions that segment properly the three organs. INET has been trained using the weighted mean as loss function as in the original work [24].

3.3 Post-processing

The predictions provided for each pixel by the neural architectures range from 0 to 1. To turn this soft classification into a binary mask, it is necessary to threshold the output at a given value. In this paper, we use the same threshold value (0.25) as in [24], where this value was fixed empirically based on a preliminary experimentation. This output does not ensure the presence of a single connected object for the heart, and two for clavicles and lungs. Therefore, a last and very simple post-processing step is considered: the largest connected object is selected for the hearts, and the two largest ones for the clavicles and lungs (notice that this post-processing step is a simplified version of the one proposed by [26], since it does not fill the holes within the object). The algorithm employed for this task was the Block-Based Decision Table algorithm [57] with 8-way connectivity. This very simple post-processing step has a minor but positive impact, as can be seen in Table 8, and it does not differ from other simple post-processing strategies employed [26, 58, 59, 60].

4 Experiments

The empirical evaluation of this paper includes two experiments. The first experiment is devoted to the study of performance, precision, robustness, and the trade-off between accuracy and memory/time consumption of the X-Net architectures and INET with both single-class and multi-class training approaches. This study is performed using a 3-fold cross validation protocol as in [24]. The goal of the second experiment is the comparison of the X-Net architectures performance (except the worst performing architectures from the latter experiment, that are excluded from the comparison) with the state-of-the-art results using a 10-fold cross validation to avoid any bias caused by the stochastic components of training a ConvNet.

It is important to highlight the computational cost of performing the experimentation following a rigorous experimental design in deep learning (cross validation). Overall, around 1608 hours (67 days) were required to perform Experiment I, and around 1840 hours (77 days) were necessary to run Experiment II. Detailed information about training times are included in Sections 4.5 and 4.6.

4.1 Data

The dataset employed in the experiments is the JSRT dataset [25]. It is the most widely used dataset in CXRs segmentation. This dataset is composed of 247 CXRs of 2048 × 2048 pixels with a grayscale depth of 12 bits. These images contain manual/ground-truth segmentations of the lungs, clavicles, and heart[26] with a resolution of 1024 × 1024 pixels, where ∼73%, ∼5%, and ∼22% of pixels belong to lungs, clavicles, and hearts, respectively.

4.2 Performance metrics

Three metrics are employed to quantitatively evaluate the quality of the segmentation results obtained: Hausdorff Distance (HD) [61], Jaccard Index (JI) [62], and DSC [50]. The HD represents a measure of the spatial distance between two sets of points: it is the largest of all distances from any point in the resulting segmentation to the closest point in the ground truth, and a value of 0 indicates perfect agreement. Meanwhile, the DSC and the JI measure set agreement: a value of 0 indicates no overlap with the ground truth, and a value of 1 indicates perfect agreement. Notice that, DSC and JI are equivalent metrics, and one can be derived from the other. Thus, for the comparison between X-Net architectures only the HD and JI will be reported. However, in order to facilitate the comparison with competitor methods (Experiment II), the DSC is also included in the tables.

Our final goal is to be able to segment CXRs in the original resolution of their ground truth segmentation (i.e. 1024 × 1024), and not in a down-sampled resolution, because up-sampling to the ground truth resolution will worsen the accuracy of the final segmentation. As consequence, all results are reported in the ground truth resolution, either if they correspond to the ConvNet output (e.g. X-Net+ and RX-Net+) or to the up-sampled version of it (INET, X-Net and RX-Net).

4.3 Experimental set-up

The first experiment involves the application of 6 deep network configurations (INET and X-Net can only run using the 256 × 256 resolution, RX-Net with 256×256 and 1024×1024 resolutions, and lastly X-Net+ and RX-Net only with the 1024 × 1024 resolution), and two training strategies (single-class and multi-class approaches). INET, X-Net and RX-Net are trained from scratch for 4000 epochs (since that was the number of epochs required by INET to converge according to [24]). X-Net+ and RX-Net+ are trained for 100 epochs using as initialization the weights of X-Net and RX-Net in the shared layers, respectively. These are tested using a 3-fold cross validation approach (as in INET [24]), where one fold is devoted to testing (33% of all available data), and each one of the remaining two folds is divided into training and validation (90% and 10%, respectively). Furthermore, the results are evaluated with and without post-processing to measure the contribution of this refinement step. To sum up, 12 deep networks are evaluated (see Table 3), rising up to 24 architectures if we include results with and without post-processing (see Table 4). The notation employed to refer to each model uses the following labeling protocol: *<Network Name>_<Single(s) or Multi(m) organ problem>_<Input/Output resolution>*. As an example, the architecture INET_m_256 corresponds to INET trained to solve the multi-class problem on 256 × 256 images.

The second experiment involves the comparison of INET and the best proposals from the latter experiment in terms of accuracy (X-Net+ for single-class and multi-class) and accuracy-memory/time balance (RX-Net+) with a 10-fold cross validation, where on each fold 80% of data are used for training, 10% for validation, and 10% for test. This allows to study more rigorously the proposals reducing possible bias, as remarked in [63], caused by the stochastic effect inherent to the training process or the effect that different training and test sets have on the final performance. The results obtained by X-Net architectures are compared among them and with the state-of-the-art (see Tables 6, 7, and 8).

Both experiments (Sections 4.5 and 4.6) include the results provided by our implementation of INET. This allows us to replicate the exact same experimental conditions in all methods and, therefore, to perform a fair comparison. The only exception is Table 8, dedicated to the comparison with the state of the art, where we show the original results reported on each paper. The difference between the results provided by our implementation of INET and the results reported in [24] is minor, as can be seen by comparing the results in the original paper with Tables 3 and 6, and can be due to several reasons: from differences in the partitions employed in the 3-fold and 10-fold, respectively; differences in the batch size employed (as theirs is not reported in the paper); or just the inherent stochastic behavior of training a network from scratch.

No data augmentation is performed and the images are zero centered, as in [24], using the mean and standard deviation of the training set. The batch size was set to 1. The optimizer is Adam (with a learning rate of 1e-5, beta1 of 0.9, and beta2 of 0.999). The outputs of lower resolution than the ground truth (i.e. 1024×1024) are scaled using a bicubic interpolation, since it showed better results than the other alternatives tested, although the gap between the best and worst interpolation was lower than 0.001 according to the JI.

All experiments have been performed on an Nvidia Titan X with 12 GBs of memory using Keras 2.1.6 with TensorFlow 1.4.1 as backend. Codes and trained architectures will be made publicly available upon acceptance of the paper.

4.4 Preliminary Experiment: Evaluating the influence of architectural changes on INET and post-processing

The purpose of the first preliminary experiment is to measure the influence of the different architectural changes introduced on INET to obtain X-Net. The results of this ablation study are shown in Table 1. The best results are obtained by instance normalization together with atrous convolution, being both sources of improvement. However, we can claim that instance normalization has a greater contribution to this improvement. Instance normalization introduces some noise into the network, helping to improve its generalization ability. We hypothesize that, since we have at our disposal a small dataset, this noise inducing process contributes to enforce regularization and, therefore, to improve the results obtained.

A second preliminary experiment was performed to measure the impact of the post-processing step (see Section 3.3). The post-processing step (see Table 2) has shown to improve the results according to both JI and HD, providing statistically significant differences according to Wilcoxon's rank sum test [64] ($9.8 \cdot 10^{-90}$ for JI;

Table 1 Summary of the preliminary experiments according to the average JI and HD of the three organs to study the influence of architectural changes on INET without post-processing. 'in' and 'ac' stand for 'instance normalization' and 'atrous convolution', respectively.

Network	**JI**			
	mean	**sd**	**min**	**max**
INET	0.885	0.012	0.685	0.944
INET + ac	0.892	0.013	0.721	0.953
INET + in	0.910	**0.007**	**0.832**	0.963
X-Net (INET + in + ac)	**0.925**	**0.007**	0.797	**0.967**
Network	**HD**			
	mean	**sd**	**min**	**max**
INET	132.37	72.38	100.33	255.27
INET + ac	128.27	68.10	99.03	239.50
INET + in	124.65	84.30	**97.54**	226.27
X-Net (INET + in + ac)	**121.68**	**62.32**	98.27	**188.73**

and 0 for HD). On average, the JI improves from 0.895 to 0.899, and the HD from 86.699 to 35.069. This simple post-processing step is important for metrics that focus on the quality of the final contours (like HD), since removing the artifacts allows for a better comparison of the error in the boundaries of the segmented organs. We want to highlight that, even if the post-processing has a positive impact on the final result, almost all X-Net architectures without post-processing yield a better performance than our implementation of INET with post-processing.

Table 2 Summary of the preliminary experiments according to the average JI and HD of the three organs to study the influence of the post-processing.

Network	Without post-processing		With post-processing	
	JI mean	HD mean	JI mean	HD mean
INET	0.876	132.522	0.883	43.143
X-Net	0.905	91.653	0.908	31.006
RX-Net	0.899	60.599	0.899	34.400

4.5 Experiment I: Comparison of X-Net architectures and INET with single-class and multi-class strategies

The results obtained for the single-class and multi-class strategies are shown in Table 3, employing JI and HD as evaluation metrics. The first conclusion worth mentioning is that single-class training strategies generally outperform multi-class strategies for CXRs segmentation. There are statistically significant differences in favor of the former with p-values, according to the Wilcoxon's rank sum test [64] of 0.02 for the JI, and $6.5 \cdot 10^{-20}$ for HD. In particular, single-class approaches obtain the best segmentation results for clavicles and lungs, while the best results on hearts are obtained by a multi-class approach. Thus, despite multi-task learning has shown useful in other problems [65, 66], its use must be studied for each particular problem. Finally, the comparison of the results of RX-Net and RX-Net+ for images of 1024 × 1024, i.e. RX-Net_m_1024 and RX-Net+_m_1024, provides support about the fact that the relation between the filter's field-of-view and the feature maps affects significantly to the performance. Since this simple

Table 3 Summary of results evaluated using JI and HD per architecture and organ. All X-Net and INET variants are included.

Network	Organ\Metric	JI					HD				
		mean	std	median	min	max	mean	std	median	min	max
INET_m_256	Clavicles	0.833	0.015	0.843	0.639	0.905	22.390	10.721	20.180	6.031	72.764
	Heart	0.869	0.024	0.894	0.511	0.955	50.245	32.378	40.464	13.867	195.971
	Lungs	0.951	0.006	0.957	0.842	0.972	56.795	40.207	42.804	13.176	229.373
INET_s_256	Clavicles	0.862	0.017	0.876	0.635	0.931	20.375	12.672	17.825	5.846	88.730
	Heart	0.866	0.032	0.896	0.350	0.961	51.971	33.839	40.764	13.403	185.146
	Lungs	0.949	0.007	0.957	0.821	0.974	49.071	37.883	35.783	11.362	207.247
X-Net_m_256	Clavicles	0.876	0.013	0.887	**0.701**	0.934	18.518	9.538	16.383	5.025	64.325
	Heart	0.890	0.014	0.905	**0.751**	0.963	38.317	18.297	34.434	11.369	**93.118**
	Lungs	0.961	0.004	0.965	0.903	0.976	36.183	25.384	27.083	10.599	**140.922**
X-Net_s_256	Clavicles	0.855	0.013	0.867	0.682	0.919	19.151	11.812	16.434	6.640	89.206
	Heart	0.889	0.017	0.905	0.654	0.969	37.501	18.953	34.505	10.295	100.593
	Lungs	0.959	0.004	0.961	0.892	0.976	36.909	29.552	**25.281**	10.343	176.254
X-Net+_m_1024	Clavicles	0.883	0.015	0.894	0.686	0.949	18.468	10.761	15.818	4.824	67.755
	Heart	**0.892**	0.014	0.903	0.735	0.965	37.732	19.318	33.318	10.889	118.317
	Lungs	**0.963**	0.004	**0.967**	**0.908**	**0.980**	38.352	27.830	27.611	10.181	144.794
X-Net+_s_1024	Clavicles	**0.885**	0.016	**0.896**	0.630	**0.953**	18.022	11.241	15.941	4.824	87.660
	Heart	0.890	0.016	**0.907**	0.706	**0.970**	**37.207**	20.542	**32.514**	**8.872**	107.929
	Lungs	**0.963**	0.004	**0.967**	0.896	**0.980**	**36.100**	29.485	26.605	**7.912**	184.837
RX-Net_m_256	Clavicles	0.860	0.017	0.874	0.661	0.929	20.047	10.479	17.202	7.066	70.187
	Heart	0.889	0.015	0.905	0.738	0.961	38.007	18.074	34.350	13.150	99.823
	Lungs	0.955	0.005	0.961	0.889	0.976	45.146	30.254	36.291	11.338	167.700
RX-Net_s_256	Clavicles	0.869	0.017	0.881	0.606	0.934	18.049	10.101	16.058	**4.667**	70.237
	Heart	0.883	0.016	0.898	0.704	0.963	40.068	19.971	36.872	11.312	105.527
	Lungs	0.959	0.004	0.963	0.899	0.976	38.694	29.625	27.335	11.105	168.505
RX-Net_m_1024	Clavicles	0.855	0.023	0.880	0.548	0.942	22.872	11.765	19.616	6.535	66.767
	Heart	0.874	0.020	0.894	0.657	0.967	46.055	26.915	39.795	13.631	155.417
	Lungs	0.951	0.007	0.959	0.823	0.976	51.204	35.061	41.329	14.524	190.915
RX-Net_s_1024	Clavicles	0.866	0.019	0.880	0.642	0.949	21.762	13.535	18.989	5.878	95.639
	Heart	0.855	0.032	0.880	0.367	0.961	49.850	31.894	41.383	12.801	182.362
	Lungs	0.953	0.006	0.961	0.869	0.976	49.069	35.060	37.430	11.656	190.086
RX-Net+_m_1024	Clavicles	0.867	0.018	0.880	0.612	0.940	19.472	9.751	16.962	7.333	60.711
	Heart	0.889	0.015	0.903	0.726	0.961	37.330	18.653	33.224	11.646	105.080
	Lungs	0.955	0.005	0.961	0.881	0.978	44.751	30.678	34.837	11.600	167.060
RX-Net+_s_1024	Clavicles	0.880	0.016	0.892	0.646	0.946	**17.728**	9.301	**15.695**	5.277	**53.971**
	Heart	0.883	0.017	0.896	0.686	0.965	40.472	21.172	35.936	11.200	119.962
	Lungs	0.961	0.004	**0.967**	0.894	0.978	38.596	29.747	26.927	10.051	169.883

post-processing has shown to be beneficial, all results of X-Net architectures and our implementation of INET include it (see Tables 3, 6, 4, and 7).

We rank the performance of the X-Net architectures, as well as our implementation of INET, in Table 4 according to JI and HD. Methods with a difference in performance smaller than 0.0025 and 5 for JI and HD, respectively, are considered equivalent. This ranking does not show the values of JI and HD, but the average position of each network for a given metric and organ. Thus, the values of the ranking goes from 1 to the number of networks, and smaller values are associated with a better performance. All our proposals outperform INET (even the reduced ones which require lower resources than INET), INET being the worst performing approach in the comparison. It is important to remember that INET is the current state-of-the-art approach in multi-class CXRs segmentation. Another important conclusion is that, generally, ground truth resolution approaches (1024 × 1024) outperform downsampled approaches. In particular, the best method in all rankings is X-Net+ in ground truth resolution using a single-class training approach (with X-Net+_m_1024 being the second best performing approach).

The time required to train INET was about 26 hours per run (i.e. 26 hours for multi-class approach and 78 hours for single-class since three networks are trained), and 9 GBs of memory are necessary (for both the single-class and multi-class configuration). X-Net requires 36 hours and 9 GBs to train, while the finetuning of X-Net+, from the X-Net weights, takes only 3 hours (for a total of 39 hours),

Table 4 Average ranking position of X-Net architectures and INET per organ and metric (JI, HD, and their average) using 3-fold cross validation [24]. Two networks are considered equal if the difference in performance between them is lower than 0.0025 for JI and 5 pixels for HD.

Network\Metric	Clavicles			Lungs			Hearts			3 organs		
	JI	HD	Aver.	JI	HD	Aver.	JI	HD	Aver.	JI	HD	Aver.
X-Net+_s_1024	**2.2**	**5.3**	**3.8**	**4.8**	**2.8**	**3.8**	**2.8**	**4.2**	**3.5**	**3.3**	**4.1**	**3.7**
X-Net+_m_1024	**2.2**	**5.3**	**3.8**	**4.8**	3.8	4.3	**2.8**	**4.2**	**3.5**	**3.3**	4.4	3.9
X-Net_m_256	3.3	**5.3**	4.3	**4.8**	**2.8**	**3.8**	**2.8**	**4.2**	**3.5**	3.7	**4.1**	3.9
RX-Net+_s_1024	2.8	**5.3**	4.1	**4.8**	5.2	5.0	6.7	5.5	6.1	4.8	5.3	5.1
X-Net_s_256	9.7	**5.3**	7.5	**4.8**	3.8	4.3	4.7	**4.2**	4.4	6.4	4.4	5.4
RX-Net_s_256	6.3	**5.3**	5.8	**4.8**	5.2	5.0	7.0	5.5	6.3	6.1	5.3	5.7
RX-Net+_m_1024	6.8	**5.3**	6.1	6.7	7.3	7.0	5.2	**4.2**	4.7	6.2	5.6	5.9
RX-Net_m_256	8.7	**5.3**	7.0	6.2	8.7	7.4	4.0	**4.2**	4.1	6.3	6.1	6.2
RX-Net_s_1024	6.8	7.7	7.3	6.2	9.5	7.8	11.7	10.3	11.0	8.2	9.2	8.7
INET_s_256	7.5	7.0	7.3	9.5	9.0	9.3	10.3	11.0	10.7	9.1	9.0	9.1
RX-Net_m_1024	9.7	11.2	10.4	11.3	10.2	10.8	9.3	9.7	9.5	10.1	10.3	10.2
INET_m_256	12.0	9.5	10.8	9.2	9.7	9.4	10.7	11.0	10.8	10.6	10.1	10.3

Table 5 Summary of the average time and memory requirements of X-Net architectures and INET with both the mono-class and multi-class approaches.

Network	**Muti-class time (h)**	**Single-class time (h)**	**GPU memory (GB)**
INET (256 × 256)	26h	26h×3=78h	9GB
INET (1024 × 1024)	Cannot be trained due to its GPU memory requirements.		
X-Net (256 × 256)	36h	36h×3=108h	9GB
X-Net (1024 × 1024)	Cannot be trained due to its GPU memory requirements.		
X-Net+ (1024 × 1024)	36h+3h =39h	(36h+3h)×3=117h	12GB
RX-Net (256 × 256)	12h	12h×3=36h	3.5 GB
RX-Net (1024 × 1024)	55h	55h×3=165h	11GB
RX-Net+ (1024 × 1024)	12h+2h =14h	(12h+2h)×3=42h	10GB

requiring almost 12GBs of GPU memory. RX-Net requires only 12 hours and 3.5 GBs with images of 256×256, and 55 hours and 11 GBs with images of 1024×1024. Meanwhile, RX-Net+ with images of 1024×1024 takes only 2 hours to be finetuned from the weight of RX-Net (256) (i.e. a total of 14 hours) and 10 GBs. Thus, RX-Net outperforms INET in accuracy but also reduces the required memory and the training time (see Table 5 for a summary of the time and memory requirement of all the architectures). Overall, around 1608 computational hours (67 days) were required to perform the 3-fold cross validation.

Given that X-Net, the proposal that is closest to INET, is better than INET (see rankings of Table 4, where X-Net_s and X-Net_m are systematically ranked above their INET counterparts), we can conclude that the modifications introduced in X-Net are responsible for such improvement. Therefore, the use of atrous convolution and instance normalization to improve the results in CXRs segmentation is highly recommended.

4.6 Experiment II: Comparison with State-of-the-art approaches

The results obtained by the best X-Net architectures employing 10-fold cross validation are shown in Table 6. All those results include post-processing. The results of INET [24] correspond to our implementation, in order to perform a comparison as rigorous as possible with the same 10-folds. The comparison of Tables 3 and 6 shows that the results obtained have not changed significantly from the 3-fold to the 10-fold cross validation protocol. X-Net architectures are robust to different initialization and training-test subsets. This is supported by the unchanged positions of the different proposals in the ranking showed in Table 7. Lastly, the Nemenyi test [67] was performed to look for statistically significant differences between the best ranked proposal, X-Net+_s_1024, and all the other networks. The test showed that there is no statistical significant difference with X-Net+_m_1024 with p-values larger than 0.1 for both and HD. Therefore, both X-Net+_s_1024 and X-Net+_m_1024 must be considered the best performing approaches. More specifically, when employing JI and DSC as evaluation metrics, X-Net+_s_1024 is better for clavicles, and X-Net+_m_1024 for lungs and hearts. However, X-Net+_s_1024 becomes also the best method in clavicles when HD is considered. The Nemenyi test finds statistically significant differences with all the other networks with a p-value always lower than $1 \cdot 10^{-06}$ for both JI and HD. In particular, for INET_m_256, it obtains a p-value of $5.2 \cdot 10^{-15}$ for JI and $1.2 \cdot 10^{-12}$ for HD.

The time required to train a fold of all architectures with the two training approaches is lower since only 3000 epochs are performed instead of the 4000 from the previous experiment, and also the number of X-Net architectures compared is lower. Nevertheless, the computational time needed to tackle this experimentation is significantly higher because a 10-fold cross validation were performed, and thus *circa* of 1840 computational hours (i.e. 77 days) were required.

X-Net+, evaluated using a 10-fold cross validation protocol, provides better results in the ground truth resolution than all the other methods in the state-of-the-art (9 competitor approaches) for clavicles and lungs. It also yields comparable results with the state-of-the-art method [38] for heart segmentation (with a difference in performance smaller than 0.01 (JI) and 0.005 (DSC)). X-Net+ also outperforms the human observer in lungs and hearts (see Table 8). Importantly, X-Net+ without post-processing yields comparable results to X-Net+ with post-processing.

5 Conclusion and future works

This paper tackles the problem of segmenting multiple organs (hearts, lungs and clavicles) in chest X-ray images using convolutional neural networks. Several new deep architectures are proposed to deal with this complex problem. First, X-Net focuses on improving the segmentation accuracy in images of 256×256 (the conventional resolution used in the literature). Second, RX-Net represents a simplification of X-Net, and it is focused on reducing the required computational resources (training memory and time) without significant loss in the original accuracy. Finally, X-Net+ and RX-Net+, are extensions of the former architectures that allow us to work with images up to 1024 × 1024, maintaining the original relation between filter's field-of-view and the feature maps, and to transfer the learning from their precedent versions (X-Net and RX-Net, respectively, for images of 1024 × 1024).

Remarkably, state-of-the-art results have been obtained. Our best performing proposal, X-Net+ for single-class segmentation, obtains better results than the state of the art with clavicles with an average error of 0.884 (JI), 0.939 (DSC), and 18.022 (HD), and lungs with 0.963 (JI), 0.981 (DSC), 36.1 (HI). The results for hearts segmentation are as good as the state of the art with 0.89 (JI), 0.942 (DSC), and 37.207 (HD). The quantitative evaluation of our X-Net architectures by means

Table 6 Summary of JI and HD results per architecture and organ employing a 10-fold cross validation protocol to the best performing architectures in Experiment I (see Section 4.5).

Network	Organ\Metric	JI					HD				
		mean	std	median	min	max	mean	std	median	min	max
INET_m_256	Clavicles	0.835	0.018	0.850	0.663	0.905	21.937	10.946	19.732	8.218	55.271
	Heart	0.850	0.033	0.883	0.546	0.944	64.605	41.183	50.472	20.807	178.663
	Lung	0.953	0.005	0.959	0.885	0.972	51.679	36.886	41.390	13.676	163.012
X-Net_m_256	Clavicles	0.848	0.017	0.862	0.667	0.910	20.507	12.043	17.734	7.519	60.869
	Heart	0.881	0.019	**0.901**	0.663	0.951	**42.179**	23.456	36.361	14.808	119.029
	Lung	0.951	0.005	0.957	0.890	0.972	47.416	35.574	35.761	13.287	170.676
X-Net_s_256	Clavicles	0.859	0.013	0.869	**0.748**	0.912	18.381	9.789	16.133	7.826	**51.733**
	Heart	0.878	0.017	0.892	**0.718**	0.955	45.603	24.703	40.355	14.842	110.086
	Lung	0.951	0.006	0.957	0.880	0.972	46.333	37.479	32.960	**11.806**	177.618
X-Net+_m_1024	Clavicles	0.874	0.018	0.889	0.678	0.940	20.361	11.891	17.336	7.587	59.583
	Heart	**0.883**	0.018	0.898	0.698	**0.959**	42.638	27.027	**35.422**	13.773	132.061
	Lung	**0.957**	0.005	**0.961**	**0.898**	**0.976**	46.477	34.094	35.544	13.321	**161.642**
X-Net+_s_1024	Clavicles	**0.883**	0.015	**0.896**	0.745	**0.944**	**18.357**	11.567	**15.795**	**6.595**	60.463
	Heart	0.878	0.018	0.896	0.714	0.957	43.671	24.269	37.844	**13.342**	**107.652**
	Lung	0.955	0.006	**0.961**	0.881	0.976	**46.248**	37.529	**32.567**	12.189	174.856
RX-Net_m_256	Clavicles	0.838	0.018	0.852	0.645	0.905	21.515	12.103	18.920	8.449	61.395
	Heart	0.876	0.019	0.896	0.682	0.951	44.508	23.747	38.192	16.442	117.139
	Lung	0.947	0.006	0.951	0.866	0.969	53.350	35.161	44.273	17.195	162.182
RX-Net_s_256	Clavicles	0.845	0.017	0.860	0.685	0.908	19.882	11.984	17.109	7.620	61.342
	Heart	0.864	0.020	0.878	0.684	0.946	50.112	30.619	42.176	16.232	147.602
	Lung	0.947	0.007	0.955	0.860	0.970	51.975	42.228	38.710	13.080	198.489
RX-Net+_m_1024	Clavicles	0.864	0.019	0.881	0.653	0.934	20.718	11.953	17.965	7.962	62.627
	Heart	0.880	0.017	0.896	0.717	0.953	43.526	23.499	37.026	15.560	111.432
	Lung	0.951	0.006	0.957	0.873	0.972	51.912	37.211	41.463	16.028	172.276
RX-Net+_s_1024	Clavicles	0.871	0.019	0.890	0.676	0.942	19.404	12.283	16.332	7.292	63.905
	Heart	0.866	0.020	0.880	0.689	0.949	50.412	31.395	43.437	16.247	144.748
	Lung	0.925	0.012	0.940	0.779	0.969	79.469	45.211	69.918	21.169	208.965

Table 7 Average ranking position of the best X-Net architectures and INET per organ and metric (JI, HD, and their average). Two networks are considered equal if the difference in performance between them is lower than 0.0025 for JI and 5 pixels for HD. A 10-fold cross validation protocol is applied to the best performing architectures in Experiment I (see Section 4.5).

Network\Metric	Clavicles			Lungs			Hearts			3 organs		
	JI	HD	Aver.	JI	HD	Aver.	JI	HD	Aver.	JI	HD	Aver.
X-Net+_s_1024	**1.8**	**3.2**	**2.5**	4.4	3.7	4.0	3.9	**3.5**	**3.7**	3.3	**3.5**	**3.4**
X-Net+_m_1024	2.5	5.4	3.9	**3.8**	**2.8**	**3.3**	**3.3**	4.1	**3.7**	**3.2**	4.1	3.6
X-Net_s_256	4.8	**3.2**	4.0	4.1	3.9	4.0	4.0	4.9	4.4	4.3	4.0	4.1
X-Net_m_256	6.5	5.4	5.9	4.1	3.4	3.8	4.0	3.7	3.8	4.8	4.2	4.5
RX-Net+_m_1024	4.1	6.4	5.2	4.5	5.6	5.0	4.5	4.4	4.4	4.3	5.4	4.9
RX-Net_s_256	6.7	5.2	5.9	5.2	5.5	5.4	6.5	6.2	6.4	6.1	5.6	5.9
RX-Net_m_256	7.8	6.0	6.9	6.3	6.2	6.2	4.9	4.3	4.6	6.3	5.5	5.9
RX-Net+_s_1024	2.8	3.4	3.1	8.7	8.5	8.6	6.4	6.0	6.2	6.0	6.0	6.0
INET_m_256	8.3	7.0	7.7	4.1	5.5	4.8	7.7	8.1	7.9	6.7	6.9	6.8

of a rigorous experimental design protocol (10-fold cross validation, rankings and statistical tests) shows the empirical advantages of employing them in this task.

Overall, the single-class training approach achieved better segmentation than the multi-class approach. This shows that multi-task learning is not always the best solution, despite its success in many other applications, and it must be analyzed for every particular problem separately. Furthermore, we have empirically shown, by comparing RX-Net and RX-Net+ for images of 1024 × 1024, that re-sizing a network to fit an input changes the relation between filter's field-of-view and the feature maps leading to a change in its behaviour. In our case, this change has significantly worsened the results of RX-Net obtaining the worst results among our proposals, meanwhile RX-Net+ has been ranked in the top 5.

Table 8 Comparison of our best X-Net-based architectures, with and without post-processing, with state-of-the-art approaches. JI and DSC represent the results reported in the trained resolution (indicated by the number in parentheses at the end of the name of the method). JI Full and DSC Full report the results in the original resolution of the segmentation mask (i.e. 1024 × 1024). DSC and DSC Full (as well as JI and JI Full) have the same value for methods trained with the original image resolution. The best results in the ground truth resolution, 1024 × 1024, are displayed in bold per organ and metric. Notice that all approaches report better results in the down-sampled resolution than in the ground truth resolution. The reason is that the resulting segmentation is evaluated more roughly, and thus we lose details and nuances. Since the segmentation results are always more precise in the ground truth resolution, we employ it as reference to highlight in bold the performance of the different algorithms under comparison. Cells containing a "—" represent either that the proposed method does not tackle the segmentation of the organ, or that the results at the original or down-sampled resolution are not reported. Values calculated from other metric, where only one of them was reported, are marked with a "*". Results without the post-processing step are also reported to allow a fair comparison with "pure" deep learning methods.

	Clavicles				**Lungs**				**Hearts**			
Method	**JI**	**JI Full**	**DSC**	**DSC Full**	**JI**	**JI Full**	**DSC**	**DSC Full**	**JI**	**JI Full**	**DSC**	**DSC Full**
Human observer [26]	—	0.896	—	0.945*	—	0.946	—	0.972*	—	0.878	—	0.935*
TVC2018 (512) [38]	—	—	—	—	—	0.951	—	0.975	—	**0.893**	—	**0.943**
WPC2018, a.k.a. LF-SegNet (224) [45]	—	—	—	—	0.951	—	0.975*	—	—	—	—	—
WPC2018-2, a.k.a. FCN (224) [48]	—	—	—	—	0.959	—	0.979*	—	—	—	—	—
JBHI2018 (256) [21]	—	—	—	—	0.952	—	0.975	—	—	—	—	—
MP2017 (256) [68]	—	—	—	—	0.955	—	0.977	—	—	—	—	—
N2018 (256) [36]	—	—	—	—	0.963	0.948	0.983	0.974	—	—	—	—
MIA2012 (256) [23]	0.860	—	0.925*	—	—	—	—	—	—	—	—	—
SCIA2017 (256) [44]	0.863	—	0.926*	—	0.959	—	0.979*	—	0.899	—	0.947*	—
TMI2018, a.k.a. INET (256) [24]	0.868	—	0.929	—	0.950	—	0.974	—	0.882	—	0.937	—
X-Net+_m_1024 without post-proc.	0.871		0.931		0.954		0.976		0.879		0.935	
X-Net+_m_1024	0.874		0.933		**0.956**		**0.978**		0.884		0.938	
X-Net+_s_1024 without post-proc.	0.880		0.936		0.954		0.976		0.863		0.927	
X-Net+_s_1024	**0.883**		**0.938**		0.955		0.977		0.879		0.935	
RX-Net+_m_1024 without post-proc.	0.859		0.924		0.948		0.973		0.876		0.934	
RX-Net+_m_1024	0.864		0.927		0.951		0.975		0.880		0.936	
RX-Net+_s_1024 without post-proc.	0.867		0.929		0.924		0.960		0.849		0.919	
RX-Net+_s_1024	0.870		0.931		0.925		0.961		0.865		0.928	

Regarding future lines of research, the first future work will be to study the capability of our proposals to be applied to other problems, such as the segmentation of different sets of organs, different datasets, and different kinds of radiographs. Second, we aim to adapt our methods to the volumetric medical image segmentation scenario following an approach similar to V-Net [69]. Third, we would like to study further network simplifications following an automatic pruning approach as in ThinNet [70]. Last, we plan to test the applicability of this segmentation framework in a challenging real world problem, such as forensic identification via comparative radiography [11], to study if the quality of the resulting segmentation is sufficient for that task.

Acknowledgements This research was supported by the Spanish *Ministerio de Economía y Competitividad* under the NEWSOCO project [grant number TIN2015-67661-P], including European Development Regional Funds (EDRF). Mr. Gómez's work was supported by Spanish MECD FPU grant [grant number FPU14/02380]. Pablo Mesejo is funded by the European Commission H2020-MSCA-IF-2016 through the Skeleton-ID Marie Curie Individual Fellowship [grant number 746592]. We acknowledge the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research.

Conflict of Interest The authors declare that they have no conflict of interest.

References

1. Staffan Sandström, Harald Ostensen, and Holger Pettersson. *The WHO manual of diagnostic imaging: radiographic technique and projections*, volume 2. World Health Organization, 2003.
2. Richard H Daffner and Matthew Hartman. *Clinical radiology: the essentials*. Lippincott Williams & Wilkins, 2013.
3. Donna-Marie Rigby and Linda Hacking. Interpreting the chest radiograph. *Anaesth Intensive Care*, 19(2):50 – 54, 2018.
4. B. Van Ginneken, B. M. Ter Haar Romeny, and M. A. Viergever. Computer-aided diagnosis in chest radiography: a survey. *IEEE Trans Med Imaging*, 20(12):1228–1241, Dec 2001.
5. NHS England. Diagnostic imaging dataset annual statistical release 2015/2016. 2016.
6. Jonathan Laserson, Christine Dan Lantsman, Michal Cohen-Sfady, Itamar Tamir, Eli Goz, Chen Brestel, Shir Bar, Maya Atar, and Eldad Elnekave. TextRay: Mining Clinical Reports to Gain a Broad Understanding of Chest X-Rays. In *MICCAI*, pages 553–561, 2018.
7. Arnold M.R. Schilham, Bram van Ginneken, and Marco Loog. A computer-aided diagnosis system for detection of lung nodules in chest radiographs with an evaluation on a public database. *Med Image Anal*, 10(2):247 – 258, 2006.
8. Ajay Mittal, Rahul Hooda, and Sanjeev Sofat. Lung field segmentation in chest radiographs: a historical review, current status, and expectations from deep learning. *IET Image Processing*, 11(11):937–952, 2017.
9. M.T. Tsakok and F.V. Gleeson. The chest radiograph in heart disease. *Medicine*, 46(8): 453 – 457, 2018.
10. Sudhir Kapoor, Akshay Tiwari, and Saurabh Kapoor. Primary tumours and tumorous lesions of clavicle. *Int Orthop*, 32(6):829, 2008.
11. Oscar Gómez, Oscar Ibáñez, Andrea Valsecchi, Oscar Cordón, and Tzipi Kahana. 3D-2D silhouette-based image registration for comparative radiography-based forensic identification. *Pattern Recognit*, 83:469 – 480, 2018.
12. Michael A Bruno, Eric A Walker, and Hani H Abujudeh. Understanding and confronting our mistakes: the epidemiology of error in radiology and strategies for error reduction. *Radiographics*, 35(6):1668–1676, 2015.
13. PJ Robinson, D Wilson, A Coral, A Murphy, and P Verow. Variation between experienced observers in the interpretation of accident and emergency radiographs. *The British journal of radiology*, 72(856):323–330, 1999.
14. Adrian Brady, Risteárd Ó Laoide, Peter McCarthy, and Ronan McDermott. Discrepancy and error in radiology: concepts, causes and consequences. *Ulster Med J*, 81(1):3, 2012.
15. Jun-Ichiro Toriwaki, Yasuhito Suenaga, Toshio Negoro, and Teruo Fukumura. Pattern recognition of chest X-ray images. *Comput Vision Graph*, 2(3):252 – 271, 1973.
16. H. Wechsler and J. Sklansky. Finding the rib cage in chest radiographs. *Pattern Recognit*, 9(1):21 – 30, 1977.
17. Ying Zhu, Simone Prummer, Peng Wang, Terrence Chen, Dorin Comaniciu, and Martin Ostermeier. Dynamic layer separation for coronary DSA and enhancement in fluoroscopic sequences. In *MICCAI*, pages 877–884, 2009.
18. Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241, 2015.
19. Bram Van Ginneken, BM Ter Haar Romeny, and Max A Viergever. Computer-aided diagnosis in chest radiography: a survey. *IEEE Trans Med Imaging*, 20(12):1228–1241, 2001.
20. Yeqin Shao, Yaozong Gao, Yanrong Guo, Yonghong Shi, Xin Yang, and Dinggang Shen. Hierarchical lung field segmentation with joint shape and appearance sparse learning. *IEEE Trans Med Imaging*, 33(9):1761–1780, 2014.
21. Wei Yang, Yunbi Liu, Liyan Lin, Zhaoqiang Yun, Zhentai Lu, Qianjin Feng, and Wufan Chen. Lung field segmentation in chest radiographs from boundary maps by a structured edge detector. *IEEE J Biomed Health Inform*, 22(3):842–851, 2018.

22. Haithem Boussaid, Iasonas Kokkinos, and Nikos Paragios. Discriminative learning of deformable contour models. In *ISBI*, pages 624–628. IEEE, 2014.
23. Laurens Hogeweg, Clara I Sánchez, Pim A de Jong, Pragnya Maduskar, and Bram van Ginneken. Clavicle segmentation in chest radiographs. *Med Image Anal*, 16(8):1490–1502, 2012.
24. A. A. Novikov, D. Lenis, D. Major, J. Hladůvka, M. Wimmer, and K. Bühler. Fully convolutional architectures for multiclass segmentation in chest radiographs. *IEEE Trans Med Imaging*, 37(8):1865–1876, Aug 2018.
25. Junji Shiraishi, Shigehiko Katsuragawa, Junpei Ikezoe, Tsuneo Matsumoto, Takeshi Kobayashi, Ken-ichi Komatsu, Mitate Matsui, Hiroshi Fujita, Yoshie Kodera, and Kunio Doi. Development of a digital image database for chest radiographs with and without a lung nodule: receiver operating characteristic analysis of radiologists' detection of pulmonary nodules. *AJR Am J Roentgenol*, 174(1):71–74, 2000.
26. Bram Van Ginneken, Mikkel B Stegmann, and Marco Loog. Segmentation of anatomical structures in chest radiographs using supervised methods: a comparative study on a public database. *Med Image Anal*, 10(1):19–40, 2006.
27. Dmitry Ulyanov, Andrea Vedaldi, and Victor S. Lempitsky. Instance normalization: The missing ingredient for fast stylization. *CoRR*, abs/1607.08022, 2016.
28. Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *CoRR*, abs/1606.00915, 2016.
29. D. J. Withey and Z. J. Koles. Medical image segmentation: Methods and software. In *NFSI-ICFBI*, pages 140–143, 2007.
30. Muhammad Waseem Khan. A survey: Image segmentation techniques. *International Journal of Future Computer and Communication*, 3(2):89, 2014.
31. Erik Smistad, Thomas L. Falch, Mohammadmehdi Bozorgi, Anne C. Elster, and Frank Lindseth. Medical image segmentation on GPUs – a comprehensive review. *Med Image Anal*, 20(1):1 – 18, 2015.
32. Pablo Mesejo, Oscar Ibáñez, Oscar Cordón, and Stefano Cagnoni. A survey on image segmentation using metaheuristic-based deformable models: state of the art and critical analysis. *Appl Soft Comput*, 44:1–29, 2016.
33. Mariano Cabezas, Arnau Oliver, Xavier Lladó, Jordi Freixenet, and Meritxell Bach Cuadra. A review of atlas-based segmentation for magnetic resonance brain images. *Comput Methods Programs Biomed*, 104(3):e158 – e177, 2011.
34. Bo Peng, Lei Zhang, and David Zhang. A survey of graph theoretical approaches to image segmentation. *Pattern Recognit*, 46(3):1020 – 1038, 2013.
35. Alberto Garcia-Garcia, Sergio Orts-Escolano, Sergiu Oprea, Victor Villena-Martinez, and Jose Garcia-Rodriguez. A review on deep learning techniques applied to semantic segmentation. *arXiv preprint arXiv:1704.06857*, 2017.
36. P. Chondro, C.-Y. Yao, S.-J. Ruan, and L.-C. Chien. Low order adaptive region growing for lung segmentation on plain chest radiographs. *Neurocomputing*, 275:1002–1011, 2018.
37. Khang Siang Tan and Nor Ashidi Mat Isa. Color image segmentation using histogram thresholding – fuzzy C-means hybrid approach. *Pattern Recognit*, 44(1):1 – 15, 2011.
38. Lei Bi, Dagan Feng, and Jinman Kim. Dual-path adversarial learning for fully convolutional network (FCN)-based medical image segmentation. *The Visual Computer*, 34(6): 1043–1052, Jun 2018.
39. Sim Kuan Goh, Hussein A Abbass, Kay Chen Tan, Abdullah Al-Mamun, Nitish Thakor, Anastasios Bezerianos, and Junhua Li. Spatio–spectral representation learning for electroencephalographic gait-pattern classification. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(9):1858–1867, 2018.
40. Shuchao Pang, Juan José del Coz, Zhezhou Yu, Oscar Luaces, and Jorge Díez. Deep learning and preference learning for object tracking: A combined approach. *Neural Processing Letters*, 47(3):859–876, 2018.
41. Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
42. Junhua Li, Zbigniew Struzik, Liqing Zhang, and Andrzej Cichocki. Feature learning from incomplete eeg with denoising autoencoder. *Neurocomputing*, 165:23–31, 2015.
43. Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*,

2017.

44. Chunliang Wang. Segmentation of multiple structures in chest radiographs using multi-task fully convolutional networks. In Puneet Sharma and Filippo Maria Bianchi, editors, *SCIA*, pages 282–289, 2017.
45. Ajay Mittal, Rahul Hooda, and Sanjeev Sofat. LF-SegNet: A fully convolutional encoder–decoder network for segmenting lung fields from chest radiographs. *Wireless Personal Communications*, 101(1):511–529, Jul 2018.
46. S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *ICML*, 1:448–456, 2015.
47. Stefan Jaeger, Sema Candemir, Sameer Antani, Yì-Xiáng J Wáng, Pu-Xuan Lu, and George Thoma. Two public chest X-ray datasets for computer-aided screening of pulmonary diseases. *Quant Imaging Med Surg*, 4(6):475, 2014.
48. Rahul Hooda, Ajay Mittal, and Sanjeev Sofat. An efficient variant of fully-convolutional network for segmenting lung fields from chest radiographs. *Wireless Personal Communications*, 101(3):1559–1579, Aug 2018.
49. Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *J Mach Learn Res*, 15(1):1929–1958, 2014.
50. Thorvald Sørensen. A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on danish commons. *Kongelige Danske Videnskabernes Selskab*, 5:1–34, 1948.
51. Misha Denil, Babak Shakibi, Laurent Dinh, Nando De Freitas, et al. Predicting parameters in deep learning. In *Advances in neural information processing systems*, pages 2148–2156, 2013.
52. S. Han, X. Liu, H. Mao, J. Pu, A. Pedram, M.A. Horowitz, and W.J. Dally. EIE: efficient inference engine on compressed deep neural network. *ISCA*, pages 243–254, 2016.
53. Yong-Deok Kim, Eunhyeok Park, Sungjoo Yoo, Taelim Choi, Lu Yang, and Dongjun Shin. Compression of deep convolutional neural networks for fast and low power mobile applications. *arXiv preprint arXiv:1511.06530*, 2015.
54. Baiying Lei, Shan Huang, Ran Li, Cheng Bian, Hang Li, Yi-Hong Chou, and Jie-Zhi Cheng. Segmentation of breast anatomy for automated whole breast ultrasound images with boundary regularized convolutional encoder–decoder network. *Neurocomputing*, 321: 178 – 186, 2018.
55. J. Liu, J. Cai, K. Chellamuthu, M. Bagheri, L. Lu, and R. M. Summers. Cascaded coarse-to-fine convolutional neural networks for pericardial effusion localization and segmentation on ct scans. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 1092–1095, April 2018.
56. Ke Gong, Xiaodan Liang, Yicheng Li, Yimin Chen, Ming Yang, and Liang Lin. Instance-level human parsing via part grouping network. *arXiv preprint arXiv:1808.00157*, 2018.
57. Costantino Grana, Daniele Borghesani, and Rita Cucchiara. Optimized block-based connected components labeling with decision trees. *IEEE Trans Image Process*, 19(6):1596–1609, 2010.
58. Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.
59. Geert Litjens, Clara I Sánchez, Nadya Timofeeva, Meyke Hermsen, Iris Nagtegaal, Iringo Kovacs, Christina Hulsbergen-Van De Kaa, Peter Bult, Bram Van Ginneken, and Jeroen Van Der Laak. Deep learning as a tool for increased accuracy and efficiency of histopathological diagnosis. *Scientific reports*, 6:26286, 2016.
60. Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 1(2):4, 2017.
61. Mario Beauchemin, Keith PB Thomson, and G Edwards. On the Hausdorff distance used for the evaluation of segmentation results. *Can J Remote Sens*, 24(1):3–8, 1998.
62. Paul Jaccard. The distribution of the flora in the alpine zone. 1. *New phytologist*, 11(2): 37–50, 1912.
63. Stéphane Lathuilière, Pablo Mesejo, Xavier Alameda-Pineda, and Radu Horaud. A comprehensive analysis of deep regression. *arXiv preprint arXiv:1803.08450*, 2018.
64. Edmund A Gehan. A generalized Wilcoxon test for comparing arbitrarily singly-censored samples. *Biometrika*, 52(1-2):203–223, 1965.

65. Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Facial landmark detection by deep multi-task learning. In *ECCV*, pages 94–108, 2014.
66. Zhizheng Wu, Cassia Valentini-Botinhao, Oliver Watts, and Simon King. Deep neural networks employing multi-task learning and stacked bottleneck features for speech synthesis. In *ICASSP*, pages 4460–4464, 2015.
67. Myles Hollander and Douglas A Wolfe. *Nonparametric statistical methods*. Wiley-Interscience, 1999.
68. J. Xiong, Y. Shao, J. Ma, Y. Ren, Q. Wang, and J. Zhao. Lung field segmentation using weighted sparse shape composition with robust initialization. *Medical Physics*, 44(11): 5916–5929, 2017.
69. Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *3DV*, pages 565–571. IEEE, 2016.
70. J. Luo, H. Zhang, H. Zhou, C. Xie, J. Wu, and W. Lin. ThiNet: Pruning CNN filters for a thinner net. *IEEE Trans Pattern Anal Mach Intell*, On press.