Automated tumor segmentation in multimodal PET / CT / MR imaging

Andrei Iantsen

DOCTORAT BIOLOGIE BRETAGNE SANTE LOIRE

UBO
Université de Bretagne Occidentale

Thèse de doctorat de

L'UNIVERSITÉ DE BRETAGNE OCCIDENTALE

École Doctorale N° 605
Biologie, Santé
Spécialité : *Analyse et Traitement de l'Information et des Images Médicales*

Par
Andrei IANTSEN

Automated Tumor Segmentation in Multimodal PET / CT / MR Imaging

Thèse présentée et soutenue à Brest, France, le 13 janvier 2022
Unité de recherche : INSERM UMR 1101 - LaTIM Laboratoire de Traitement de l'Information Médicale

Rapporteurs avant soutenance :

Adrien DEPEURSINGE	Professeur, HES-SO, Centre Hospitalier Universitaire Vaudois, Suisse
Su RUAN	Professeure, Université de Rouen Normandie, France

Composition du Jury :

Président :	Roland HUSTINX	Professeur Université de Liège, CHU de Liège, Belgique
Examinateurs :	Adrien DEPEURSINGE	Professeur, HES-SO, Centre Hospitalier Universitaire Vaudois, Suisse
	Su RUAN	Professeure, Université de Rouen Normandie, France
	Dimitris VISVIKIS	Directeur de recherche, INSERM UMR 1101 - LaTIM, France
Dir. de thèse :	Mathieu HATT	Directeur de recherche, INSERM UMR 1101 - LaTIM, France

Summary

Over a half-century ago, one of the central characters in a series of novels by English science-fiction writer Arthur C. Clarke was not a human being but a computer. As conceived by the writer, this machine had intelligence and was able to speak, recognize faces, process natural languages and play chess, among other things. What was considered a sheer fiction at that time is now becoming a reality due to rapid advances in science and digital technologies. Data is increasingly seen by many as "the new oil", and artificial intelligence (AI) is an engine of productivity and economic growth. Machines performing human-like cognitive processes, such as learning, understanding, reasoning and interacting, deeply transform the way in which modern societies live and work. In healthcare, AI-based visual recognition systems have the potential to revolutionize the disease diagnosis process by helping radiologists interpret medical images and accelerating reading time. In order to be employed for clinical decision support, such mechanisms must be *reliable*, *accurate* and *integrated* into the workflow. The first two requirements, i.e., the *reliability* and *accuracy* of AI-based solutions, are central research subjects of this thesis, examined in the context of automated tumor segmentation in multimodal medical imaging. The integration aspects are beyond the scope of this work and only briefly covered in Chapter 8.

Chapter 1 introduces a statistical learning framework developed by V. Vapnik, which is de facto the foundation of modern data analysis. Starting with a description of the key postulates of supervised learning, this chapter formulates a task of empirical risk minimization, describes associated pitfalls, e.g., overfitting, and advocates the use of techniques such as regularization, cross-validation and ensembling in order to reduce the generalization error and obtain its reliable estimate. The last part of the chapter describes a special type of ensemble technique known as stacking and its close connection with feedforward neural networks.

Chapter 2 is devoted to neural networks that form the core of deep learning. First, it introduces two basic models, namely linear regression and logistic regression, and standard statistical techniques for estimating their parameters on training

data, i.e., least squares and maximum likelihood estimation, respectively. Then, it is demonstrated that a feedforward neural network is nothing more than a series of logistic regression models stacked one on top of another. Next, convolutional neural networks (CNNs) are presented as a specialized kind of networks for working with grid-like data, such as images. Finally, approximation properties of different types of neural networks are provided as a family of theorems stating that (almost) any function can be approximated arbitrarily well by a neural network of a certain type.

Chapter 3 gives a general overview of architectures used in various medical image segmentation tasks. It is shown that the vast majority of existing CNNs are based on U-Net, and often involve just minor modifications proposed for specific tasks and/or datasets. The main sources of variation in architectural design are summarised in this chapter.

The following chapters are based on research projects carried out during my PhD studies. Chapter 4 repeats the content of the article on cervical cancer segmentation in PET imaging. One purpose of that study was to propose a U-Net based model for fully automated delineation of 3D functional primary tumor volumes in the specific context of cervical cancer, where a pathological uptake of interest is located close to a physiological one, corresponding to the bladder. A secondary objective was to train the network on reliable ground-truth segmentation masks obtained through accurate and robust PET semi-automated segmentation instead of manual delineation. A final objective was to train and evaluate the model performance under standard clinical imaging conditions, considering a multicenter patient cohort without any prior standardization in data acquisition and/or image reconstruction processes. As a results, the designed model performed well for this task, and it was demonstrated on a dataset comprised of 232 patients from five institutions. A versatile pipeline was designed, including appropriate data preprocessing and augmentation steps, design of the model architecture beyond the standard U-Net model, and an optimized training procedure. All experiments were conducted in the multicenter context to imitate a typical clinical scenario, in which this task can arise.

An automated approach to head and neck primary tumor segmentation in combined PET/CT images in the context of the MICCAI 2020 HECKTOR challenge is described in Chapter 5. A part of this chapter repeats exactly our previously published findings. However, a large amount of supplementary information was included relying on papers published after the end of the challenge. The key ingredient of the solution is a new computational unit, called Squeeze-and-Excitation Normalization, that was proposed by our team to supplement the vanilla U-Net model. Using a train-

ing set of 201 patients from four medical centers for model development, the designed method obtained the best results among all participating teams on an independent test set and won first prize in the contest. Both development and testing were done in the multicenter fashion, and the model predictions were accurate and robust without any center-specific standardization. Moreover, an estimate for inter-observer agreement between four human experts was considerably worse than the model results, demonstrating the high potential of CNN-based models in this task.

Chapter 6 describes a CNN-based method for brain tumor segmentation in multisequence MRI scans. This method was applied to delineate three different glioma sub-regions in the context of the MICCAI 2020 BraTS challenge. Apart from a few minor modifications, the model described in this chapter is identical to the one presented in Chapter 5. Moreover, pipelines (i.e., data preprocessing, augmentation, training procedures, etc.) in both chapters vary insignificantly and mainly because of the difference in input image modalities. Nonetheless, the described approach obtained highly competitive results in the BraTS contest as well, and finished fourth in the final ranking, essentially without any task-specific adjustments.

The primary objective of Chapter 7 is to investigate the feasibility of achieving fully automated detection and segmentation of lymphoma and sarcoidosis lesions in PET/CT images by applying the original U-Net model off-the-shelf, i.e., without any task- and data-specific adjustments. This aim was chosen to check if high-quality results can be achieved by using the original model with the fixed architecture and fine-tuning solely the pipeline components. The analyses was carried out on a retrospective dataset consisting of 419 patients with biopsy-proven diagnoses, using two complementary groups of metrics. As a result, the trained model obtained good average accuracy for all metrics in the segmentation task. On the other hand, the detection performance varied significantly depending on the chosen detection criteria.

Conclusions and future research perspectives are discussed in Chapter 8.

Contents

List of Tables

List of Figures

Chapter 1

Statistical Learning Framework

1.1 Learning Task

Broadly speaking, the term "learning" can be explained as a process of improving on specific tasks with experience. The central subject of this process is a *learner*, or a *learning system*, which is typically considered to be a machine or a computer program. In *supervised learning* tasks, there exists a *domain set* X, wherein each element is represented by a *feature vector*, i.e., $x = (x^1, x^2, \ldots, x^p)$ for all $x \in X$, and an associated *label set* Y. A *training set* $S = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$ is a sequence of N labeled domain points, often called *training examples*, that represent experience available to the learning system. The purpose of learning on the training set S is to obtain a *predictor* $h : X \to Y$, also called a *hypothesis* or a *model*, that can be used to label new domain points. In other words, having the training set S, the leaner makes a reasonable guess about the data-generating process and selects the best candidate h from a set of available hypotheses H.

Underlying assumptions about the data-generating process have to be made. Define a joint probability distribution $\mathcal{D}$ over domain points and labels, $X \times Y$. Suppose that all elements in the training set S are *independent and identically distributed* (i.i.d) according to the probability distribution $\mathcal{D}$ that is unknown to the learner, i.e. $(x, y) \sim \mathcal{D}$ for any $(x, y) \in S$.

For any training example (x, y), an error between the *ground-truth* label y and the predicted label $h(x)$ can be measured with a *loss function* $\ell : Y \times Y \to \mathbb{R}_+$. Define a *generalization error*, or a *true risk*, of a hypothesis h to be

$$L_{\mathcal{D}}(h) \stackrel{\text{def}}{=\!=} \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\ell(y, h(x))\right] \tag{1.1}$$

that is the expected value of the loss function measured with respect to the probability distribution $\mathcal{D}$. Since the distribution $\mathcal{D}$ is unknown, the generalization error is not directly available to the learner. An estimate for the generalization error can be a *training error*, also called an *empirical risk*, that is calculated directly on the training set S:

$$L_S(h) \stackrel{\text{def}}{=\!=} \frac{1}{N} \sum_{i=1}^{N} \ell(y_i, h(x_i)) \,, \tag{1.2}$$

where the subscript S indicates that the error is computed on the training set S. Therefore, the task of *Empirical Risk Minimization* (ERM) is to find a hypothesis $h_s^* \in H$ using the training set S to minimize the empirical risk:

$$h_s^* = \underset{h \in H}{\operatorname{argmin}} \frac{1}{N} \sum_{i=1}^{N} \ell(y_i, h(x_i)) \,. \tag{1.3}$$

It is worth to note that in the field of machine learning, the learning process is often considered from the perspective of function approximation. Often, this assumes the existence of some "correct" labeling function, $f : X \to Y$, such that $f(x) = y + \varepsilon$ for all $(x, y) \in X \times Y$, where ε represents the random error independent of the input, with zero mean and finite variance, i.e., $\mathbb{E}[\varepsilon] = 0$ and $Var(\varepsilon) = \sigma^2$. The error term is used to introduce uncertainty associated with the data-generating process (e.g., unsystematic errors in labels). Thus, from this perspective, the learning task is to obtain the best approximation (in terms of some criterion) of the target function f using any available hypothesis from H and the training set S.

1.2 Regression and Classification

There exist a wide variety of supervised learning tasks. However, only two of them, namely regression and classification, are central in the context of this work. Both tasks have a training set $S = \{(x_1, y_1), \ldots, (x_N, y_N)\}$ of N labeled domain points.

In *regression tasks*, each ground-truth label $y \in Y$ is represented by a real number. For example, one can consider survival analysis that plays crucial role in clinical research. The main objective is to predict patient survival, i.e., time to death, based on some clinical information about patients provided as a set of features. In this case, the quality of a prediction $h(x)$ for a data point (x, y) can be measured by a *squared*

loss:

$$\ell_{sq}(y, h(x)) \stackrel{\text{def}}{=\!=} (y - h(x))^2 \,. \tag{1.4}$$

The empirical risk with the squared loss is often referred to as a *mean squared error* (MSE). For the training set S with N data points, it is written as

$$L_S^{MSE}(h) \stackrel{\text{def}}{=\!=} \frac{1}{N} \sum_{i=1}^{N} (y_i - h(x_i))^2 \,. \tag{1.5}$$

The problem of survival prediction can be reformulated as a *classification task*. Suppose the label set Y is a finite set of *categories*, or *classes*, so that each patient belongs to one of them. In the aforementioned example, one can stratify all patients into short-, mid- and long-survivors depending on their survival time that corresponds to *multiclass classification*. The most intuitive way to assess the prediction rule, which in this context is commonly called a *classifier*, is to rely on a *0-1 loss*:

$$\ell_{0-1}(y, h(x)) \stackrel{\text{def}}{=\!=} \begin{cases} 0 & \text{if } \; h(x) = y \\ 1 & \text{if } \; h(x) \neq y \end{cases} \tag{1.6}$$

that indicates if the prediction is incorrect. Thus, the empirical risk corresponds to a proportion of mislabeled examples, i.e., an *error rate* (ER):

$$L_S^{ER}(h) \stackrel{\text{def}}{=\!=} \frac{1}{N} \sum_{i=1}^{N} \ell_{0-1}(y_i, h(x_i)) \,. \tag{1.7}$$

Under certain conditions, the regression problem has a single analytical solution (see details in Section 2.1). On the other hand, the classification task with the 0-1 loss is typically NP-hard, and therefore computationally intractable [235]. To circumvent this problem, one common approach is to replace the 0-1 loss with a *surrogate loss*, which is often its convex upper bound. This convex relaxation allows to derive a computationally efficient solution relying on numerical optimization methods.

1.3 Gradient Descent

Among numerical methods suitable for different learning tasks, most effective modern algorithms are built on *gradient descent*, also known as *steepest descent*. Suppose each hypothesis $h \in H$ is uniquely defined by a vector of parameters $\mathbf{w}$. Therefore, the

ERM task can be written as

$$\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmin}}\ L_S(\mathbf{w}) \tag{1.8}$$

for the training set S. Gradient descent iteratively performs small updates of the parameters $\mathbf{w}$ in the direction of the negative gradient, so that

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \alpha \boldsymbol{\nabla} L_S(\mathbf{w}^{(t)})\,, \tag{1.9}$$

where $t = 0, 1, \ldots$ denotes the iteration step and $\alpha > 0$ is the hyperparameter, known as a *learning rate*, that controls the step size. This process usually starts with the weights $\mathbf{w}^{(0)}$ that are initialized randomly or following some task-specific initialization scheme [33, 66, 90]. The gradient is re-evaluated for the new weight vector after each update in order to move the value of the training error towards the local minimum. The training procedure continuous while a *stopping criterion*, e.g., the maximum number of training iterations, is not satisfied.

Unfortunately, a single weight update requires to compute the gradient with respect to the whole training set S, and it can be inefficient or even intractable in practice. Therefore, instead of using the entire training set S, the gradient of the training error in Equation (1.9) can be replaced with its estimate computed on a *mini-batch* B to obtain *mini-batch gradient descent*:

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \alpha \boldsymbol{\nabla} L_B(\mathbf{w}^{(t)})\,, \tag{1.10}$$

where B is a subset of training examples from S. The use of small mini-batches significantly reduces computational burden of training complex models and allows to move rapidly through the weight space in large-scale problems. The specific case with $|B| = 1$ is known as *stochastic gradient descent* that provides weight updates on the basis of the loss function measured for a single example.

The *batch gradient descent*, i.e., the method that operates on the entire training set, is shown to monotonically converge to the local minimum, whereas stochastic gradient descent typically oscillates in some area around it due to noise in the gradient estimates. The level of noise can be dampen by applying *gradient descent with momentum* that performs weight updates using gradients from previous iterations so it can considerably accelerate convergence [176, 186]. This method is typically has a form of

$$\mathbf{v}^{(t+1)} = \beta \mathbf{v}^{(t)} - \alpha \boldsymbol{\nabla} L_B(\mathbf{w}^{(t)})\,, \tag{1.11}$$

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} + \mathbf{v}^{(t+1)}\,, \tag{1.12}$$

where $\alpha > 0$ is the learning rate, $\beta \in [0, 1]$ is another hyperparameter, called a *momentum coefficient*, that limits the contribution of previous gradients to the current update. The case of $\beta = 0$ represents the mini-batch gradient descent method, when the weight update at each iteration is solely determined by the corresponding gradient. Denote the weight update $\alpha L_B(\mathbf{w}^{(t)})$ at the iteration t by $\Delta^{(t)}$ and rewrite Equation (1.12) as follows:

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \Delta^{(t)} - \beta\Delta^{(t-1)} - \beta^2\Delta^{(t-2)} - \beta^3\Delta^{(t-3)} - \dots \tag{1.13}$$

It is evident that the momentum method in fact takes into account the exponential moving average of the previous updates so it allows to reduce oscillations and speed up training. In practical application, the momentum coefficient β is commonly set between 0.9 and 0.99 and rarely fine-tuned.

The learning rate in the equations above is treated as a parameter that remains constant in the course of training. However, it was shown that in many cases the learning rate adjustments can be beneficial for improving convergence. It is usually considered good practice to start with the high learning rate and gradually reduce it over training iterations to not overshoot a local minimum. One option is to apply a *learning rate schedule* that basically determines the learning rate as a function of the training step t. For example, a learning rate decay can be achieved with the schedule of the form

$$\alpha^{(t)} = \frac{1}{1 + \gamma t}\alpha^{(0)}\,, \tag{1.14}$$

where $\alpha^{(0)}$ is the initial learning rate, and $\gamma \in [0, 1]$ is a *decay parameter*. More sophisticated strategies are based on the idea of cyclical learning rates that often yield better results in fewer iterations and without a need to carefully fine-tune the learning rate [146, 199, 200]. As an alternative, one can rely on adaptive learning rate methods, such as RMSProp [211] and Adam [122], that implement individual learning rates for different parameters. Nowadays, these methods become a default option in many practical application due to fast convergence and robustness to random parameter initialization [185], that can be detrimental for previously mentioned gradient descent methods. On the other hand, a number of studies demonstrated that the adaptive methods tend to have worse generalization performance than gradient descent with

momentum [121, 226, 241]. Therefore, the choice of the optimization method can be considered, to a certain extent, as another hyperparameter depending on the task at hand.

1.4 Risk Upper Bound

The empirical risk provides an estimate of the generalization error evaluated on the training set. Since training examples are sampled randomly, the empirical risk is also a random variable that depends on a number of factors, namely the probability distribution $\mathcal{D}$, the size of the training set N and the hypothesis space H. Statistical learning theory [219] allows to compute analytic approximations, or bounds, to the generalization error based on these factors and the empirical risk. In case of a finite hypothesis space H, the following theorem can be applied [48].

Theorem 1.4.1 *For any probability distribution $\mathcal{D}$ and any training set S with a size of N drawn from $\mathcal{D}$, the probability that the absolute difference between the empirical risk and the generalization error will be greater than any $\varepsilon > 0$, in the worst case, is upper bounded as follows:*

$$P\left(\max_h |L_S(h) - L_{\mathcal{D}}(h)| > \varepsilon\right) \leq 2|H|e^{-2N\varepsilon^2}, \tag{1.15}$$

where $|H|$ is a size of the hypothesis space H.

In the equation 1.15, the difference between the empirical risk and the generalization error is often called a *generalization gap*. The aforementioned theorem states that this gap increases with the size of the hypothesis space but decreases with the size of the training set. The key intuition behind this theorem suggests that the large hypothesis space and relatively small training set will likely result in a hypothesis with the low train error just by chance, whereas the generalization gap will be large. This phenomenon is crucial in practice and known as *overfitting*.

Note that Theorem 1.4.1 considers only situations with the finite hypothesis set H. Therefore, it cannot be applied, for example, to parametric hypotheses with real-valued parameters, since the hypothesis space is infinite in this case. The statistical learning theory [219] provides methods to derive upper bounds for the generalization gap in the infinite case from the *Vapnik-Chervonenkis (VC) dimension* that expresses the *capacity* of prediction rules in the hypothesis space. See [218] for details. However, the VC dimension is hard, often unfeasible, to compute for various classes of hypotheses and the upper bounds are typically very loose [160, p. 210].

1.5 Risk Decomposition

Since overfitting tends to occur in situations with the "unlimited" hypothesis space, a common solution is to supplement the ERM task with additional restrictions that depend on the capacity of the hypotheses. These restrictions might reflect some *prior knowledge* about the learning task. For example, one can consider only the class of linear hypotheses, especially if the training set is small and linear approximation of the underlying distribution seems reasonable. However, it cannot guarantee that the hypothesis with the best achievable generalization error is still available to the learner. This dilemma is often called a *bias-variance tradeoff* and can be illustrated by separating the generalization error into three components [16, 48].

Define the *Bayes error* $B_{\mathcal{D}}$ for a given probability distribution $\mathcal{D}$ over $X \times Y$ as the infimum of the generalization error that can be achieved by any possible hypothesis h:

$$B_{\mathcal{D}} \stackrel{\text{def}}{=\!=} \inf_{\text{any } h} L_{\mathcal{D}}(h)\,. \tag{1.16}$$

This is the *irreducible generalization error* for the distribution $\mathcal{D}$ that reflects the possible non-determinism in the data-generating process and captures many real-world problems. For example, the Bayes error arises if some identical data points have different labels or some labels are assigned with errors.

Ideally, the hypothesis space H, provided to the learning system, should be rich enough to contain hypotheses with the smallest achievable error, i.e., the Bayes error. Nevertheless, choosing the richest hypothesis space - the class of all functions over the given domain - is not reasonable due to overfitting. Therefore, restricting the hypothesis space to a specific class of functions might increase the generalization error. Define an *approximation error* as the minimal possible increment over the Bayes error in the hypothesis space H:

$$L^{app}_{\mathcal{D},H} \stackrel{\text{def}}{=\!=} \min_{h \in H} L_{\mathcal{D}} - B_{\mathcal{D}}\,. \tag{1.17}$$

The approximation error is independent of the training set and only determined by the distribution $\mathcal{D}$ and chosen hypothesis space H. This error can be decreased only by enlarging the hypothesis space.

Introduce an *estimation error* caused by the fact that the learner attempts to select the best available hypothesis using the empirical risk as an estimate of the generalization error. The estimation error implies that even if the hypothesis with

the satisfying approximation error is available to the learner, it might not correctly choose it based on the available training set S. Suppose h_s^* denotes the ERM solution (see Equation 1.3), then the estimation error can be written as follows:

$$L_{\mathcal{D},S}^{est} \stackrel{\text{def}}{=\!=} L_{\mathcal{D}}(h_s^*) - \min_{h \in H} L_{\mathcal{D}}\,. \tag{1.18}$$

The estimation error decreases if the learner chooses better candidates among available hypotheses. Therefore, this error is influenced by the training set and its size as well as the size and complexity of the hypothesis space.

To sum up, the generalization error of the ERM hypothesis h_s^*, obtained on the training set S and the hypotheses space H, can be represented as follows:

$$L_{\mathcal{D}}(h_s^*) = B_{\mathcal{D}} + L_{\mathcal{D},H}^{app} + L_{\mathcal{D},S}^{est}\,. \tag{1.19}$$

Since the underlying distribution $\mathcal{D}$ is unknown, none of these errors could be evaluated. However, this decomposition clearly demonstrates the bias-variance tradeoff. On one hand, choosing a very rich set H decreases the approximation error, but at the same time it might increase the estimation error because of overfitting. This situation is often described as *"low bias - high variance"*. On the other hand, choosing a very small H reduces the estimation error but might increase the approximation error or, in other words, might lead to underfitting (*"high bias - low variance"*).

1.6 Regularization

Minimizing the empirical risk might results in overfitting, if available hypotheses have an excessive capacity for a given task. Naturally, one can supplement the ERM task (see Equation 1.3) with a complexity term to penalize more complex hypotheses. Typically, if a hypothesis is represented by a function, or a model, with a set of parameters $\mathbf{w}$, then the total number of these parameters can be treated as the model complexity. Define a *regularizer* as a functional $\mathcal{R} : H \to \mathbb{R}$ that measures the complexity of hypotheses. Supplement the ERM objective function (see Equation 1.3) with the regularizer in order to obtain the task of *Regularized Risk Minimization (RRM)*:

$$h_s^* = \operatorname*{argmin}_{h \in H} \left(\frac{1}{N} \sum_{i=1}^{N} \ell(y_i, h(x_i)) + \mathcal{R}(h) \right) \tag{1.20}$$

for some loss function ℓ and the training set S. Various methods built on the RRM

principle are widely known in statistics. They mainly differ in terms of loss functions, regularizers and assumptions about the underlying data distribution. Among them are Mallow's statistic [65, 152], the Akaike information criterion (AIC) [2], the Bayesian information criterion (BIC) [194] and the minimum description length (MDL) [76, pp. 235]. These approaches quantify the model complexity using the number of parameters which might be suboptimal. As an alternative, there exists the general measure of complexity, known as the VC dimension, which is the essence of the *Structural Risk Minimization (SRM)* principle, developed by Vapnik [219] in the context of the statistical learning theory.

In addition, there is a vast array of regularizers to induct some prior belief into the task at hand. Tikhonov, a pioneer of the regularization theory, introduced regularization as a method to *stabilize* the learning task [86, pp. 315]. Intuitively, a hypothesis is considered *stable*, if small changes in the input do not effect the output much. In other words, similar examples tend to have similar labels. Tikhonov demonstrated that this property can be achieved by restricting parameters, or weights, of the hypotheses. Suppose each hypothesis $h(\mathbf{w}) \in H$ is uniquely determined by a vector of real-valued weights, $\mathbf{w} = (w_1, w_2, \ldots, w_p)^\top$, then *Tikhonov's regularizer* is defined as:

$$\mathcal{R}(h) = \lambda \|\mathbf{w}\|_2^2 \,, \tag{1.21}$$

where $\lambda \geq 0$ and $\|\mathbf{w}\|_2 = \sqrt{\sum_{i=1}^{p} w_i^2}$. The regularization parameter λ controls the importance of the regularization term. If $\lambda = 0$, the objective function corresponds to the unconstrained case, i.e., the ERM task. The limiting case, $\lambda \to \infty$, leads to the most stable, yet trivial, hypothesis with all parameters $\mathbf{w}$ equal to zero. Therefore, the regularization parameter λ should be chosen between these extreme cases. Since the L_2-norm is used in Equation 1.21, this is often, especially in the field of machine learning, referred to as *L_2-regularization*.

One could choose the L_1-norm (without squaring) instead, that leads to *L_1-regularization*, which encourages a sparse solution:

$$\mathcal{R}(h) = \lambda \|\mathbf{w}\|_1 \,, \tag{1.22}$$

where $\|\mathbf{w}\|_1 = \sum_{i=1}^{p} |w_i|$. In particular, this type of regularization is useful in case of over-parameterization in order to drive some redundant parameters to zero and produce a parsimonious model [192].

In fact, both aforementioned regularizers can be applied simultaneously that leads

to the *elastic net regularization* [248]:

$$\mathcal{R}(h) = \alpha\|\mathbf{w}\|_2^2 + (1-\alpha)\|\mathbf{w}\|_1 \tag{1.23}$$

with the mixing parameter α, that $0 \leq \alpha \leq 1$, to balance different types of penalty.

In general, regularization techniques should be considered as certain constraints imposed on the hypothesis space. But beyond explicit requirements for the model parameters, more subtle regularization methods can be employed. Suppose, for example, that the solution of the learning task must be robust to a certain level of noise that might be present in data. This condition can be satisfied by augmenting available data examples with random noise before (or during) training, which serves as a specific form of *data augmentation*. Enlarging the training sample with augmented data points, will likely compel the learner to focus solely on a set of noise-tolerant hypotheses, that corresponds to regularization. In addition, Bishop [20] showed that training with noise is equivalent to Tikhonov regularization under certain conditions.

Apart from the orthodox methods described above, plenty of diverse, task-specific regularization techniques has been proposed [29, 164, 203].

1.7 Validation and Testing

Since the empirical risk is computed on the training set, it will be henceforth referred to as a *training error* for convenience. As mentioned earlier, the use of this measurement to estimate the generalization error typically leads to over-optimistic results caused by overfitting. Although some methods (e.g., Mallow's statistic, AIC and BIC) provide an analytical approximation of the generalization error, they are applicable only in certain settings. These approaches rely heavily on assumptions about the data-generating process and are primarily used in statistics.

In machine learning, on the other hand, the predictive accuracy is the number one priority that leads to the use of more complex and less interpretable models. In general, an estimate of the predictive performance is obtained using a *test set*. Based on the fact that the generalization error refers to the expected error on previously unseen data, one can exclude some examples from the available training set and use them afterwards to compute a *test error*. Given that the test sample is not used by the learner to choose the best hypothesis, it allows to alleviate the problem of overfitting, and thus entails a more reliable estimate of the generalization error. Note that the test set is considered to be unavailable to the learner during training and

can be used only once, solely for *performance evaluation*.

Any learning task includes two groups of parameters. The first group relates to hypotheses available to the learner. These parameters are determined, or fitted, by the learner in the course of training using some learning procedure. The second group, often referred to as *hyperparameters*, is external to the learner and must be specified before learning. For example, in the case of Tikhonov regularization (Equation 1.21), the regularization penalty is controlled by the hyperparameter λ. As another example, one can consider a set of features that characterizes each data example, since multiple ways of feature extraction might be applied depending on the learning task. It is not allowed to use the test set for hyperparameter tuning, otherwise the test error would be the biased estimate of the generalization error. Instead, a part of data, called a *validation set*, could be held out from training and testing, and used specifically to validate the choice of hyperparameters.

It is implied that partitioning of the available data into three sets, or folds, is equivalent to sampling three independent sets according to the distribution $\mathcal{D}$. In practice, however, random partitioning might results in non-representative folds, especially if the number of available examples is small, that is often termed as *selection bias*. If there is reason to believe that some features are more important for prediction than others, it is usually a good idea to use them for stratified sampling to address selection bias. For example, in some medical applications, patient age is crucial for survival prediction, and therefore different age groups should be evenly distributed between the folds. In some applications, selection bias can be tested by training a classifier that tries to identify which fold a data example comes from. Ideally, predictions of the classifier should not be significantly different from random guessing. Otherwise, it might be a sign of selection bias.

If available data is scarce, the test error might have a relatively high variance. In practice, it is often solved by applying *cross-validation*[1], which is probably the simplest and most widely used method for estimating the generalization error nowadays. Typically, in the case of *K-fold cross-validation*, the data is randomly split into K roughly equal-sized parts. Then $K-1$ parts of the data are used for building the model, whereas the kth part is held out to calculate the prediction error. This *fit-predict cycle* is repeated for $k = 1, 2, \ldots, K$ to obtain K estimates of the generalization error. The final estimate is received by taking the arithmetic mean of the K

[1]This procedure is conventionally referred to as "cross-validation", whereas in fact it corresponds to testing the model on different folds multiple times. Hence, "cross-testing" seems to be a more precise term. However, throughout this work, the conventional term is used to avoid any possible misunderstanding.

estimates. If $K = N$, i.e. each fold consists of a single data example, this method is known as *leave-one-out cross-validation* (LOOCV). In this case, the computational burden might be considerable, since it requires N repeats of the fit-predict cycle.

It is known that K-fold cross-validation provides an unbiased estimate of the generalization error, however, its variance might be large [18, 56]. This variance was estimated in a number of studies [49, 162] under certain assumptions. Nevertheless, Bengio and Grandvalet [19] demonstrated that exists no universal (valid under all distributions) unbiased estimator of the variance of K-fold cross-validation. In addition, the optimal number of folds depends on the learning task. For example, Kohavi [124] showed that in some situations large values of K might lead to an increase in variance, and therefore moderate values (10–20) are more preferable.

Surprisingly, the validation procedure had been ignored in statistical research for decades and became widely known only in the late 1970s owing to Stone [205], Mosteller and Tukey [159], and Allen [3]. The dominant idea at the time was that the data-generating process was known in advance. It was typically assumed to be a linear function with a set of parameters that were to be estimated from the training set. As a result, this simplification led to irrelevant theory, questionable conclusions, and kept statisticians from working on a large range of problems. According to Breiman [26],

> "Given that the data is generated this way, elegant tests of hypotheses, confidence intervals, distributions of the residual sum-of-squares and asymptotics can be derived. This made the model attractive in terms of the mathematics involved. This theory was used both by academic statisticians and others to derive significance levels for coefficients on the basis of model, with little consideration as to whether the data on hand could have been generated by a linear model. Hundreds, perhaps thousands of articles were published claiming proof of something or other because the coefficient was significant at the 5% level [without knowing whether the model fits the data]."

1.8 Ensemble Methods

Given a training set S, a learning system outputs a hypothesis $h \in H$ following some learning procedure. The training set S is randomly sampled from a distribution $\mathcal{D}$, therefore the resulting hypothesis h is also a random variable with some unknown

properties. Repeating the learning procedure with M independent training set samples, $S_1, S_2, \ldots, S_M$, will result in different hypotheses, $h_1, h_2, \ldots, h_M$, that in general provide different predictions for the same input x. Instead of just selecting a single hypothesis, it is often beneficial to use all of them as an *ensemble*, or a *committee*, by combining individual predictions in some way. In regression tasks [2], for example, the average could be used to obtain the ensemble prediction for a domain point x:

$$h^{\text{ens}}(x) \stackrel{\text{def}}{=\!=} \frac{1}{M} \sum_{i=1}^{M} h_i(x)\,. \tag{1.24}$$

The task of constructing "good" ensembles was one of the most active areas of research in supervised learning. As a result, it was demonstrated that ensembles are often much more accurate than individual members that make them up [50, 166]. A necessary and sufficient condition for an ensemble to outperform each individual method is if it consists of accurate and diverse hypotheses. In this context, it implies that the individual hypotheses make different, yet reliable predictions with a low correlation between their errors.

Dietterich [50] suggested three fundamental reasons for the superior performance of ensembles. From a *statistical* perspective, a learning system can be view as searching a hypothesis space H to identify the best-performing hypothesis. Without enough data, the learning system can find many different hypotheses, that all provide the same accuracy on the training set. Therefore, by constructing an ensemble out of all of these accurate hypotheses, the learner can reduce the chance of selecting the wrong one. A *computational* reason relates to situations when the best hypothesis is selected by performing some form of local search that may get stuck in local optima. In particular, this includes all sorts of neural networks that employ gradient descent to determine the best configuration, i.e. weights of the network, that minimizes a loss on the training set. Even in cases where there is enough training data, gradient descent methods lead to different results depending on weight initialization. Hence, an ensemble of neural networks trained with different starting configurations tends to provide better results [73]. The last reason is *representational* meaning that in some cases the data-generating process cannot be sufficiently approximated by any single hypothesis in H. That is why forming the ensemble of hypotheses drawn from H might expand the set of representable functions and subsequently improve generalization.

[2] One can apply majority voting as a strategy to combine independent classifiers in classification problems.

There is a number of general ways to build ensembles in practice. The first group of methods is focused on manipulating the available training data to generate multiple hypotheses. A naive approach suggests for this purpose to run the learning algorithm several times with different training samples drawn from the data distribution $\mathcal{D}$. This technique is especially effective for unstable algorithms, such as decision trees and neural networks, that are prone to overfitting. In practical applications, however, there is merely one training sample available for learning. For this reason, one can rely on the *bootstrap*, which is a method widely used in statistics to mimic the random sampling process [55, 57]. Given the training set $S = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$ comprised of N data points, the *bootstrap sample* $S^* = \{(x_1^*, y_1^*), (x_2^*, y_2^*), \ldots, (x_N^*, y_N^*)\}$ is generated by sampling the same number of random points with replacement from the training set S. Therefore, the ensemble prediction can be written as

$$h^{\text{bag}}(x) \stackrel{\text{def}}{=\!=} \frac{1}{M^*} \sum_{i=1}^{M^*} h_i(x)\,, \tag{1.25}$$

where each hypothesis h_i is received on its individual bootstrap sample S_i^*, and M^* denotes the ensemble size. The procedure of building ensembles on bootstrap samples was initially proposed by Breiman [24], who called it "bootstrap aggregating" and introduced the acronym *bagging*. Breiman [24] provided both theoretical and experimental evidence that bagging can significantly improve unstable hypotheses. In other words, it reduces the estimation error without affecting the approximation error (see Section 1.5).

In bagging, each bootstrap sample is expected to have nearly 36% duplicates caused by sampling with replacement [24], so it might entail too high correlation between all ensemble members. Parmanto et al. [172] proposed to use an alternative approach inspired by cross-validation. This method first splits all data into several folds and then build hypotheses on different combinations of training folds, which exactly repeats the cross-validation procedure. Finally, the received hypotheses are combined in ensemble. This method, originally called a *cross-validation committee*, can provide less correlated hypotheses, compared to the bootstrap, and is commonly used nowadays.

In fact, the correlation between methods can also be reduced by increasing any sort of diversity between them. Ho [96] proposed a *random subspace method* that constructs an ensemble of decision trees trained with different subsets of features. Breiman [25] applied this idea in combination with bagging that led to *random forests*. In case of neural networks, various types of data augmentation, regularization tech-

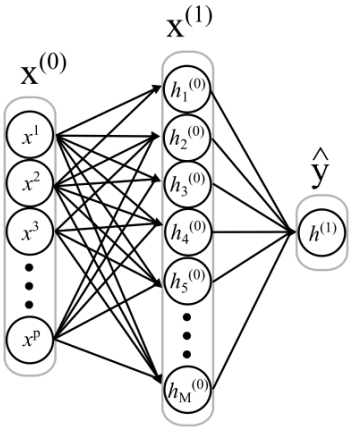


Figure 1.1: Two-level stacking with M base learners and a single super learner. Arrows denote the information flow from the original features (the level 0 space) through the level 1 space to the output.

niques as well as training procedures can be employed for this purpose.

In addition to averaging, individual hypotheses can be combined in many other ways to form ensembles. In general, it can be considered as another learning problem that requires to find a "higher-level" model that uses predictions of the ensemble members as its own features. This idea was originally proposed by Wolpert [227], who called it "stacked generalization", although nowadays the term "stacking" is more common. In this approach, cross-validation is first applied to learn a set of individual hypotheses. Wolpert refers to them as "level 0 generalizers", since they are built on the original features inhabited in a "level 0 space". Denote them by $\{h_1^{(0)}, h_2^{(0)}, \ldots, h_M^{(0)}\}$ with a superscript indicating the level. Then, their individual predictions are generated for each domain point x represented by a feature vector $\mathbf{x}^{(0)}$ in the training set in order to get the new feature representation in a "level 1 space", $\mathbf{x}^{(1)} = (h_1^{(0)}(\mathbf{x}^{(0)}), h_2^{(0)}(\mathbf{x}^{(0)}), \ldots, h_M^{(0)}(\mathbf{x}^{(0)}))$. Finally, a "level 1 generalizer", sometimes called a *super learner*, is trained to combine the lower-level generalizers and to get the final prediction $\hat{y}$ for the domain point x:

$$\hat{y} = h^{(1)}(\mathbf{x}^{(1)}) = h^{(1)}\left(h_1^{(0)}(\mathbf{x}^{(0)}), h_2^{(0)}(\mathbf{x}^{(0)}), \ldots, h_M^{(0)}(\mathbf{x}^{(0)})\right) . \tag{1.26}$$

In the original paper, Wolpert considered only the case of two-level stacking (see Figure 1.1), although a larger number of levels can be employed. It is noteworthy that the idea behind stacking shares a certain similarity to feedforward neural networks. Both neural networks and stacked ensembles aim to approximate the data-generating process by a composition of some base learners arranged in layers or levels,

respectively. However, while feedforward neural networks consist solely of basic computational units of the same type, i.e, neurons, stacking is typically used with diverse learners of any complexity. Also, the stacked learners are to be fitted independently on sophisticated data splits, whereas a feedforward neural network can be trained at once that is often described as *end-to-end training*.

Chapter 2

Methods

2.1 Linear Regression and Least Squares

In a *regression task*, a training set S consists of labeled domain points

$$\{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$$

such that $x_i \in \mathbb{R}^p$ and $y_i \in \mathbb{R}$. A typical choice to evaluate the quality of a hypothesis $h \in H$ that predicts a real-valued output $h(x)$ is to compute the *expected squared error*:

$$L_{\mathcal{D}}(h) \stackrel{\text{def}}{=\!=} \mathbb{E}_{(x,y)\sim\mathcal{D}}(y - h(x))^2 \,. \tag{2.1}$$

In the form of ERM, this equation for the training set S can be written as

$$L_s(h) \stackrel{\text{def}}{=\!=} \frac{1}{N}\sum_{i=1}^{N}(y_i - h(x_i))^2 \,. \tag{2.2}$$

In statistics and many other quantitative fields, *linear regression models* play a crucial role due to their simplicity and interpretability. The linear regression is based on the assumption that underlying relationship in data can be modeled or reasonably approximated by an affine function. Therefore, any hypothesis $h \in H$ is represented as an affine function parameterized by a vector of weights $\mathbf{w} = (w_0, w_1, w_2, \ldots, w_p)^\top$, such that

$$h(x) = w_0 + \sum_{j=1}^{p} w_j x^j \,, \tag{2.3}$$

where the parameter w_0 is called an *intercept* or a *bias*. The algorithm of solving the ERM problem for the hypothesis class of affine predictors with the respect to the expected squared loss is termed *least squares*. It represents the following optimization task:

$$\min_{\mathbf{w}} L(\mathbf{w}) \stackrel{\text{def}}{=\!=} \sum_{i=1}^{N} (y_i - w_0 - \sum_{j=1}^{p} w_j x_i^j)^2 \,. \tag{2.4}$$

Denote by $\mathbf{y} = (y_1, y_2, \ldots, y_N)^\top$ the N-vector of outputs in the training set, and similarly let $\mathbf{X}$ be the $N \times (p+1)$ matrix of features, where the *i-th* row corresponds to features of the *i-th* training example, i.e., $\mathbf{X}_i = (1, x_i^1, x_i^2, \ldots, x_i^p)$. The first component of $\mathbf{X}_i$ is a dummy variable equal to 1 in order to omit the bias term in the equations. Rewrite Equation 2.4 in matrix notation as

$$L(\mathbf{w}) = (\mathbf{y} - \mathbf{X}\mathbf{w})^\top (\mathbf{y} - \mathbf{X}\mathbf{w}) \,. \tag{2.5}$$

This is a quadratic function with $p + 1$ parameters. Differentiating with respect to $\mathbf{w}$ will give

$$\frac{\partial L(\mathbf{w})}{\partial \mathbf{w}} = -2\mathbf{X}^\top (\mathbf{y} - \mathbf{X}\mathbf{w}) \,, \tag{2.6}$$

$$\frac{\partial^2 L(\mathbf{w})}{\partial \mathbf{w} \partial \mathbf{w}^\top} = 2\mathbf{X}^\top \mathbf{X} \,. \tag{2.7}$$

Assuming that $\mathbf{X}$ has full column rank, and hence $\mathbf{X}^\top \mathbf{X}$ is positive definite, will allow to set the first derivative (Equation 2.6) to zero:

$$\mathbf{X}^\top (\mathbf{y} - \mathbf{X}\mathbf{w}) = 0 \tag{2.8}$$

in order to obtain the unique closed-form solution (the least squares estimate):

$$\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y} \,. \tag{2.9}$$

Therefore, the prediction $\hat{\mathbf{y}}$ for the training set $\mathbf{X}$ is written as

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y} \,. \tag{2.10}$$

If some features are linearly dependent, which is known in statistics as *multi-collinearity*, then $\mathbf{X}$ is not of full rank and $\mathbf{X}^\top \mathbf{X}$ cannot be inverted. In this case, the least squares estimate is not uniquely defined and can be derived from the Moore-

Penrose pseudo-inverse of the matrix $\mathbf{X}^\top\mathbf{X}$ (see [21, pp. 142]).

Linear regression is a convenient, yet quite limited method built on the assumption that dependencies in data can be captured by a linear (affine) function. However, linear regression can be made to model nonlinear relationships by transforming the input features with some fixed nonlinear functions, which is equivalent to generating a new feature space. Denote by $\phi_k(x) : \mathbb{R}^p \to \mathbb{R}$ the kth transformation of x, where $k = 1, \ldots, m$. Then, the regression model has a form

$$h(x) = w_0 + \sum_{k=1}^{m} w_k \phi_k(x)\,, \tag{2.11}$$

where $\phi_k(x)$ are known as *basis functions*. Since this method typically leads to an increased number of features, it is referred to as *basis function expansion* [21, pp. 139]. The use of basis functions allows the regression model to be a nonlinear function of the input x, while the model in fact remains linear in the parameters $\mathbf{w}$. This linearity in the parameters greatly simplifies the analyses of this class of models.

There are many possible choices for the basis functions, for example, a *sigmoidal basis function* of the form

$$\phi_\sigma(x^j) = \sigma\left(\frac{x^j - \mu}{s}\right)\,, \tag{2.12}$$

where μ and σ is a pair of hyperparameters, while $\sigma(t)$ is a *logistic function*, also called a *sigmoid*, defined by

$$\sigma(t) \stackrel{\text{def}}{=\!=} \frac{1}{1 + e^{-t}}\,. \tag{2.13}$$

Note that $\phi_\sigma(x^j)$ is a transformation defined for a single feature x^j in the feature vector x. Likewise, this function is widely used in feedforward neural networks as an activation function for nonlinear transformation of outputs in intermediate layers.

2.2 Logistic Regression and Maximum Likelihood Estimation

If all examples in the training set $S = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$ have binary labels instead of real numbers, i.e., $y_i \in \{0, 1\}$ for any $i = 1, \ldots, N$, this learning task is an instance of a *binary classification problem*. The solution of this type of problems can be obtained by adapting the linear regression method. Suppose there exists a

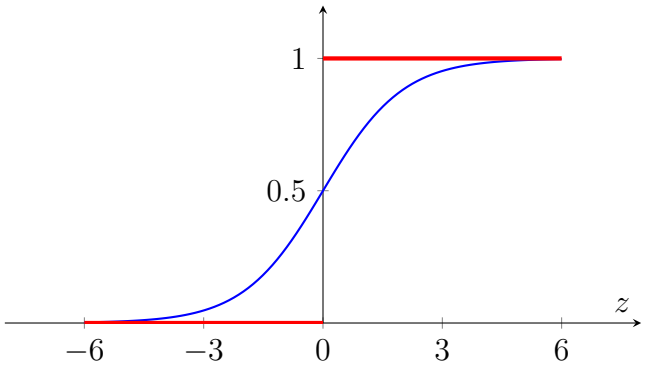


Figure 2.1: Unit step function (red) and sigmoid (blue).

linear decision boundary $z(x)$ in the feature space X, which is defined as before by a parameter vector $\mathbf{w} = (w_0, w_1, w_2, \ldots, w_p)^\top$, i.e.

$$z(x) = w_0 + \sum_{j=1}^{p} w_j x^j \,, \tag{2.14}$$

such that it separates data points into two classes. Therefore, a prediction rule can be expressed as a *unit step function*:

$$s(x) = \begin{cases} 1 & \text{if } \; z(x) > 0 \,, \\ 0 & \text{if } \; z(x) \leq 0 \,. \end{cases} \tag{2.15}$$

Unfortunately, the unit step function has its own drawbacks. The output of this function remains constant for all data points lying on the same side of the decision boundary, even if these points are significantly distinct. From the perspective of optimization, this function is not suitable as it is discontinuous at $z = 0$ and has zero gradient everywhere else. Therefore, it is better to replace the unit step function $s(z)$ with the sigmoid $\sigma(z)$, defined by Equation (2.13), that serves as its smooth, monotonic approximation (see Figure 2.1). The received model is commonly referred to as *logistic regression*.

The sigmoid function squashes the linear prediction $z(x)$ into the $(0, 1)$ interval that is regarded as the conditional probability of getting the positive class, namely

$$\begin{aligned} P(y = 1|x) &= \sigma(z) \,, \\ P(y = 0|x) &= 1 - P(y = 1|x) = 1 - \sigma(z) \,. \end{aligned} \tag{2.16}$$

These equations can be combined as

$$\begin{aligned} P(y|x) &= [P(y=1|x)]^{y} \cdot [1 - P(y=1|x)]^{1-y} \,, \\ &= \sigma(z)^{y} \cdot [1 - \sigma(z)]^{1-y} \,. \end{aligned} \tag{2.17}$$

Based on the assumption that all data points are i.i.d, the joint probability distribution of the training sample S (called a *likelihood function*) can be written as

$$\begin{aligned} L(\mathbf{w}) = P(y_1, \ldots, y_N | x_1, \ldots, x_N) &= \prod_{i=1}^{N} P(y_i|x_i) \,, \\ &= \prod_{i=1}^{N} \sigma(z_i)^{y_i} \cdot [1 - \sigma(z_i)]^{1-y_i} \,, \\ &= \prod_{i=1}^{N} \sigma(\mathbf{w}^\top \mathbf{x}_i)^{y_i} \cdot \big[1 - \sigma(\mathbf{w}^\top \mathbf{x}_i)\big]^{1-y_i} \,, \end{aligned} \tag{2.18}$$

where $z_i = w_0 + \sum_{j=1}^{p} w_j x_i^j = \mathbf{w}^\top \mathbf{x}_i$ and $\mathbf{x}_i^0 = 1$ is the dummy component to incorporate the bias. The principle idea behind *maximum likelihood estimation* (MLE) is to find the parameters $\mathbf{w}$ that maximize the likelihood function of the training set S:

$$\underset{\mathbf{w}}{\text{argmax}} \prod_{i=1}^{N} \sigma(\mathbf{w}^\top \mathbf{x}_i)^{y_i} \cdot \big[1 - \sigma(\mathbf{w}^\top \mathbf{x}_i)\big]^{1-y_i} \,. \tag{2.19}$$

It is convenient to apply the natural logarithm that leads to maximization of the *log-likelihood*:

$$\underset{\mathbf{w}}{\text{argmax}} \sum_{i=1}^{N} \big[y_i \log \sigma(\mathbf{w}^\top \mathbf{x}_i) + (1 - y_i) \log\big(1 - \sigma(\mathbf{w}^\top \mathbf{x}_i)\big)\big] \,. \tag{2.20}$$

The task (2.20) is equivalent to minimizing the *negative log-likelihood*, which is also known as *binary cross-entropy*:

$$\underset{\mathbf{w}}{\text{argmin}} \; -\sum_{i=1}^{N} \big[y_i \log \sigma(\mathbf{w}^\top \mathbf{x}_i) + (1 - y_i) \log\big(1 - \sigma(\mathbf{w}^\top \mathbf{x}_i)\big)\big] \,. \tag{2.21}$$

In contrast to linear regression, the optimization task (2.21) does not have a closed-form solution and is typically solved by numerical optimization methods, e.g.,

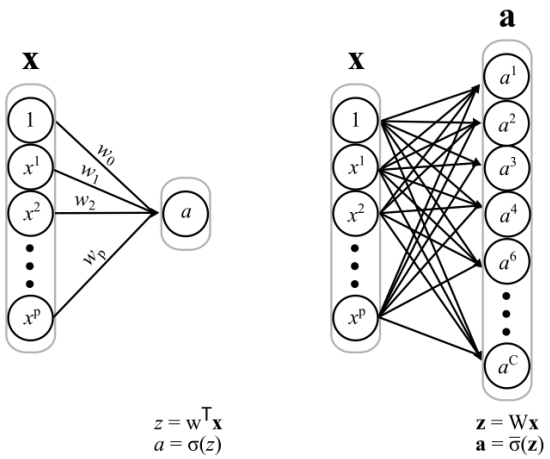


Figure 2.2: Binary (left) and multiclass (right) logistic regression models.

gradient descent and its variations (see Section 1.3 for details). Denoting the objective function (2.21), i.e., binary cross entropy, as $L^{CE}(\mathbf{w})$, it is required to compute the gradient $\boldsymbol{\nabla} L^{CE}(\mathbf{w})$. One can demonstrate [21, pp. 140] that the partial derivative of the objective function with respect to each training parameter w_j with $j = 0, \ldots, p$ has a form of

$$\frac{\partial L^{CE}(\mathbf{w})}{\partial w_j} = \sum_{i=1}^{N} \left[y_i - \sigma(\mathbf{w}^\top \mathbf{x}_i) \right] x_i^j \tag{2.22}$$

and can be used at each step of gradient descent.

The logistic regression approach can be generalized to multiclass classification, where all data labels belongs to a finite set of integer categories, i.e., $y \in \{1, \ldots, C\}$ for all $(x, y) \in S$. It usually requires to apply *one-hot encoding*:

$$t(y) = \underbrace{(0, \ldots, 1, \ldots, 0)}_{k\text{th coordinate is } 1}, \quad \text{if} \quad y = k \tag{2.23}$$

in order to transform the labels into indicator vectors. Instead of the parameter vector $\mathbf{w}$, used in the binary classification, the $C \times (p+1)$ *weight matrix* $\mathbf{W}$ is applied to map the input features $\mathbf{x} \in \mathbb{R}^{(p+1)}$ into $\mathbf{z} \in \mathbb{R}^C$, namely

$$\mathbf{z} = \mathbf{W}\mathbf{x}\,, \tag{2.24}$$

$$\begin{bmatrix} z^1 \\ z^2 \\ \vdots \\ z^C \end{bmatrix} = \begin{bmatrix} — & \mathbf{w}_1^\top & — \\ — & \mathbf{w}_2^\top & — \\ & \vdots & \\ — & \mathbf{w}_C^\top & — \end{bmatrix} \times \begin{bmatrix} x^0 \\ x^1 \\ \vdots \\ x^p \end{bmatrix} \tag{2.25}$$

where the vector $\mathbf{z}$ is commonly referred to as *logits* (see Figure 2.2). Similarly to binary classification (see Equation 2.16), the vector of logits $\mathbf{z}$ can be transformed to represent a probability distribution over the output classes as

$$P(y = k|x) = \overline{\sigma}^k(\mathbf{z})\,, \tag{2.26}$$

where each component of the vector function $\overline{\sigma}$, conventionally called a *softmax function*, is defined as

$$\overline{\sigma}^k(\mathbf{z}) = \frac{exp(z^k)}{\sum_{c=1}^{C} exp(z^c)}\,, \quad \text{for all} k \in \{1, \ldots, C\}\,. \tag{2.27}$$

Finally, *multiclass cross-entropy* for the training set with N data examples can be expressed as

$$L^{CE}(\mathbf{W}) = -\sum_{i=1}^{N}\sum_{c=1}^{C} \mathbf{t}_i^c \log(\overline{\sigma}^c(\mathbf{z}_i)) = -\sum_{i=1}^{N} \mathbf{t}_i^\top \log(\overline{\sigma}(\mathbf{z}_i))\,, \tag{2.28}$$

where $\mathbf{z}_i = \mathbf{W}\mathbf{x}_i$, the vector $\mathbf{t}_i$ is the one-hot representation (see Equation 2.23) of the label y_i for all $i = 1, \ldots, N$, and $\log(\overline{\sigma}(\mathbf{z}_i))$ corresponds to the natural logarithm applied element-wise to the softmax vector $\overline{\sigma}(\mathbf{z}_i)$. As before, gradient descent methods can be employed to minimize the objective function (2.28) with respect to the weight matrix $\mathbf{W}$ (see details in [21, pp. 209]). Note that the logistic regression and linear regression models have convex objective functions that guarantee the convergence of gradient-based optimization algorithms to a global minimum [68, pp. 169].

The multiclass logistic regression model obtains a prediction $\hat{y}$ for an example x in three steps:

$$\begin{aligned} \mathbf{z} &= \mathbf{W}\mathbf{x}\,, \\ \mathbf{a} &= \overline{\sigma}(\mathbf{z})\,, \\ \hat{y} &= \text{argmax}(\mathbf{a})\,. \end{aligned} \tag{2.29}$$

Since the softmax function outputs the probability distribution over the target labels, the argmax function is applied at the last step to get the predicted label. In

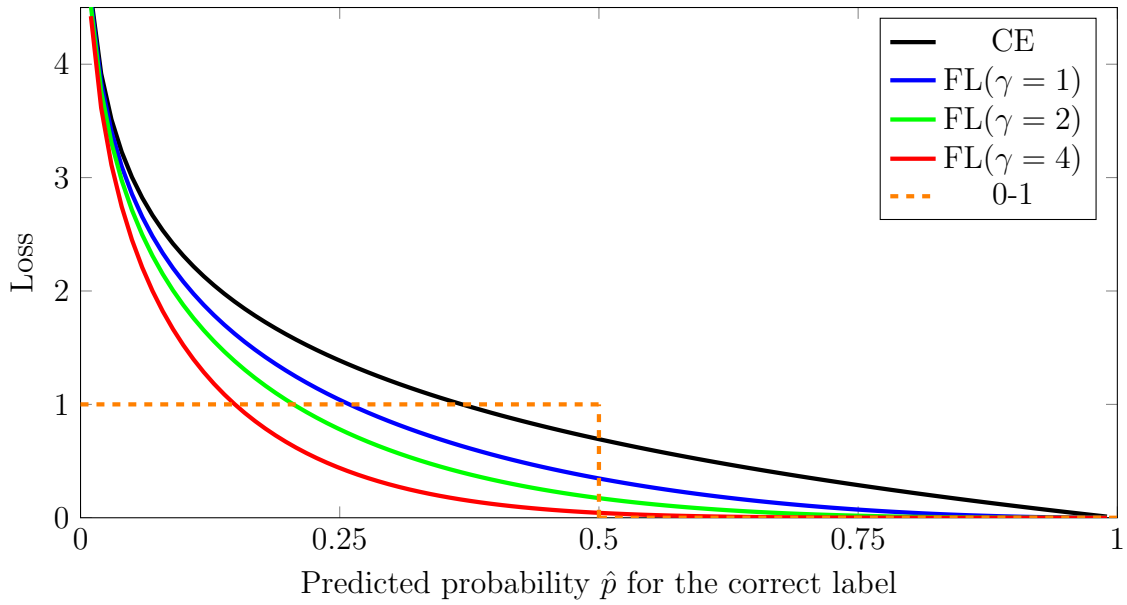


Figure 2.3: Cross-entropy loss (black), 0-1 loss (orange) and Focal loss with different values of the focusing parameter γ, that controls the loss contribution from easy examples.

practice, the softmax output is often referred to as "soft labels", since it is associated with the probability, whereas the one-hot label representation is called "hard labels".

Suppose $N = 1$ in Equation 2.28 and denote by $\hat{p}$ the predicted probability corresponding to the correct label, i.e., $\hat{p} = \mathbf{t}^\top \overline{\sigma}(\mathbf{z})$. Thus, a *cross-entropy loss* is defined as

$$\ell_{ce} \stackrel{\text{def}}{=\!=} -\log(\hat{p})\,. \tag{2.30}$$

The cross-entropy loss is a surrogate loss function suitable for gradient descent and is, in fact, a smooth approximation of the 0-1 loss (see Figure 2.3), introduced in Section 1.2. Sometimes, however, the use of cross-entropy might results in overfitting, since always $\hat{p} \to 1$ in the course of training. This phenomenon is also known as "overconfidence" [210] that can be detrimental, especially if the training set includes incorrectly labeled examples, or in case of a high class imbalance in data. This effect can be mitigated by applying different oversampling and undersampling techniques [95] or by relying on early-stopping [232]. An elegant alternative, called a *focal loss*, was proposed by Lin et al. [139] that added a modulating factor $(1 - \hat{p})$ to the cross entropy loss:

$$\ell_{fl} \stackrel{\text{def}}{=\!=} -(1 - \hat{p})^\gamma \log(\hat{p}) \tag{2.31}$$

with a focusing hyperparameter $\gamma \geq 0$. This hyperparameter smoothly down-weights the loss on easy examples and thus focus training on hard false negative examples (see Figure 2.3).

2.3 Feedforward Neural Networks

Both linear regression and logistic regression are appealing methods mainly due to the fact that they are compact and can be trained efficiently, either by applying the closed-form solution or relying on gradient descent. On the other hand, the capacity of these methods is strictly limited to linear (affine) functions that makes them incapable of learning more complex, nonlinear interactions between input features. One way to increase the capacity of linear models is to transform input features using nonlinear basis function (see Section 2.1) that is in fact equivalent to generating a new feature representation. Another option is to rely on handcrafted feature extraction techniques that typically require special knowledge and expertise in a particular domain. This approach dominated in speech recognition and computer vision before the rise of deep learning, and nowadays this principle is still at the core of radiomics analysis [1, 46, 204]. Finally, the strategy of deep learning aims at extracting a good feature representation directly from data in a fully automated manner.

A *feedforward neural network* is a quintessential deep learning model that comprises multiple layers of logistic regression models stacked one on top of another. This formulation clearly emphasizes the direct link between feedforward neural networks and the concept of stacking, presented in Section 1.8. More formally, a feedforward neural network F with d layers is a composite function of the form

$$F(\mathbf{x}) \stackrel{\text{def}}{=\!=} f^{(d)}(f^{(d-1)}(\dots(f^{(1)}(\mathbf{x}))\dots)), \tag{2.32}$$

where $\mathbf{x}$ is the input feature vector, and $f^{(i)}$ is the vector function of the *i-th* layer consisting of two transformations, namely

$$\begin{aligned} \mathbf{z}^{(i)} &= \mathbf{W}^{(i)}\mathbf{a}^{(i-1)}, \\ \mathbf{a}^{(i)} &= g(\mathbf{z}^{(i)}). \end{aligned} \tag{2.33}$$

The weight matrix $\mathbf{W}^{(i)}$ defines the affine transformation performed in the *i-th* layer, and g is the nonlinear basis function that is typically called an *activation function*, or a *nonlinearity*, in the field of deep learning. The output of the *i-th* layer,

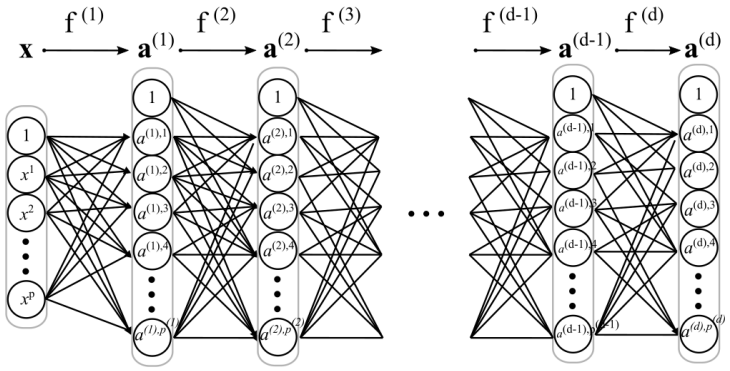


Figure 2.4: Feedforward neural network with d fully connected layers.

$\mathbf{a}^{(i)}$, is not directly observable and therefore this layer is named a *hidden layer*. The overall number of layers, d, is commonly called a model *depth*, whereas the length of the vector $\mathbf{a}^{(\mathbf{i})}$, denoted by $p^{(i)}+1$, is referred to as a *width* of the *i-th* layer (with the first component always equal to 1 to omit the bias term in Equations 2.33). Thus, each weight matrix $\mathbf{W}^{(i)}$ has a shape of $(p^{(i)}+1)\times(p^{(i-1)}+1)$.

The activation function of the last layer depends on the task at hand. However, all other layers typically have the same activation function applied element-wise. It is important to use nonlinear activation functions, otherwise the whole model collapses into a linear transformation, since the composition of successive linear transformations is itself a linear transformation. Early, models used to rely on the sigmoid (see Equation 2.13), whereas a *rectified linear unit* (ReLU) is a default recommendation for modern neural networks. The ReLU activation is defined as

$$g(z) \overset{\text{def}}{=\!=} \max(0, z)\,, \tag{2.34}$$

which means that this is a piecewise function with two linear segments (see Figure 2.5). It has been demonstrated [163] that this activation function has a number of advantages over the sigmoid so it leads to a better performance of gradient-based optimization methods. Since the activation function is applied element-wise, individual components of the vector $\mathbf{a}^{(i)}$ can be considered as basic computational units, *neurons*, that act in parallel. In fact, each neuron corresponds to a logistic regression model with some activation function g that uses the entire output of the previous layer as its own feature vector. Due to the full connectivity of adjacent layers, they are often termed "fully connected layers".

During inference, a feedworward neural network uses an input $\mathbf{x}$ to produce an output $\hat{y}$. In this case, information flows forward through the network, that is called

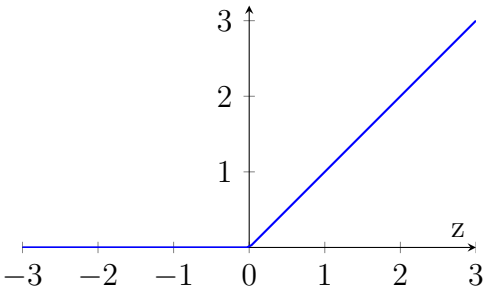


Figure 2.5: Rectified linear unit (ReLU). The recommended activation function for most modern feedforward neural networks.

forward propagation. In the course of training, forward propagation is followed by computing a value of the loss function on a mini-batch of training examples. Then, a *back-propagation* algorithm, or *backprop*, is used to send information backward through the network in order to compute the gradient of the loss function with respect to the model parameters. Therefore, back-propagation is referred to the method for computing the gradient, which can be used in combination with gradient descent methods to fit the model parameters. Apart from feedforward neural networks, the back-propagation algorithm can be applied to many other learning tasks that involve computing other derivatives as well. See details regarding the back-propagation algorithm in Rumelhart et al. [187].

2.4 Convolutional Neural Network

Feedforward neural networks have an input x in a form of a real-valued feature vector. In many applications, however, it is required for a model to process data with a grid-like structure, e.g., images, represented as multidimensional arrays, also called *tensors*. A naive approach is to flatten an n-dimensional tensor into a vector and use it as input. Since, each single neuron in feedforward neural networks interacts with the whole output of the previous layer (see Equations 2.33), the naive approach will be feasible in terms of memory requirements and statistical efficiency, only if the input images are relatively small. A better alternative is to rely on a specialized kind of neural networks for working with grid-like data, which is known as *convolutional neural networks* or CNNs.

CNNs usually have *sparse interactions*, also called *sparse connectivity*, meaning that these models focus on extracting local features that depend only on small regions of the input image. This is built on the assumption that nearby image pixels are

more strongly correlated than more distant pixels. Therefore, CNNs replace most of matrix multiplications (see Equation 2.33) with an operation called *convolution*. Suppose that I and K are two-dimensional (2D) tensors, then a (discrete) convolution[1] (denoted with an asterisk) is defined as

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(i+m, j+n)K(m,n) + b\,, \tag{2.35}$$

where (i,j) - is a pair of coordinates indicating a particular element of a tensor, the tensor I is called an *input*, whereas the tensor K is named a *kernel* or *filter*, and $b \in \mathbb{R}$ is the bias. The output tensor S is in fact a 2D feature grid that is often referred to as a *feature map*.

The filter K is typically a small tensor, e.g., 3×3, of parameters that are to be fitted in the course of training so that it makes CNNs dramatically more efficient than matrix multiplications. Since this kernel is used at every position of the input to generate the feature map, it is often referred to as *parameter sharing*.

Convolution with a single kernel can only extract one kind of features at many locations. Therefore, it is required to apply multiple different kernels and stack their feature maps along a so-called *channel dimension*, which will technically lead to the 3D output tensor. In practice, however, the channel dimension is considered independently from the spacial dimensions and can be conveniently regarded as a feature vector associated with a particular spatial point of the input tensor. In addition, the term "convolution" is typically referred to multi-channel convolution, rather than the operation presented in Equation 2.35. More precisely, *multi-channel convolution* is defined as follows:

$$S(c,i,j) = (I * K)(c,i,j) = \sum_p \sum_m \sum_n I(c+p, i+m, j+n)K(p,m,n) + b\,, \tag{2.36}$$

where c denotes the index along the channel dimension. Likewise, this operation can be defined for 3D tensors by adding an extra spatial dimension. If it is required to keep the spatial size of the input unchanged, it can be achieved by applying convolution with *padding* that first adds extra, empty pixels around the boundary of the input. The default convolutional transformation implies to slide one element at a time over all filter locations. This behavior can be modified with another parameter,

[1] In signal processing, on the other hand, this operation is usually referred to as "cross-correlation".

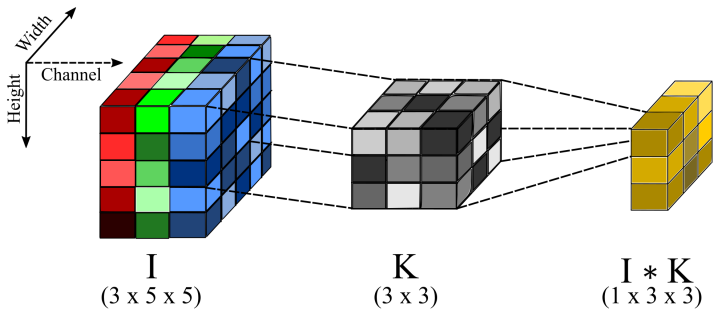


Figure 2.6: Multi-channel convolution with the kernel K of the size 3×3 applied to the input tensor I of the spatial size 5×5 with 3 input channels (the bias term b is omitted for simplicity).

called a *stride*. Similarly to feedforward neural networks, outputs of convolutional transformations are followed by an element-wise nonlinearity, such as ReLU. Overall, a set of convolutional transformations applied to the same input forms a *convolutional layer*.

The spacial size of feature maps can be decreased with the use of *pooling*. This operation replaces values at a certain spatial location with a summary statistic computed within its neighborhood. For example, *max pooling* returns a maximum value within each rectangular region of the predefined size. Likewise, *average pooling* computes a mean inside these regions (see Scherer et al. [191]). As an alternative to pooling, the output size can be reduced by applying a convolutional layer with a larger stride. To increase the spatial size, one can use upsampling with different forms of interpolation or a specialized operation known as *transposed convolution*.

Since deep learning models are commonly fitted employing gradient descent methods with a single learning rate for all parameters (see Section 1.3), input features should be normalized, i.e., adjusted to a common scale, to improve convergence. For example, one can apply *Z-score normalization*, which is widely used in statistics and defined as

$$x'_i = \frac{x_i - \mathbb{E}[X_i]}{\sqrt{Var(X_i)}}, \tag{2.37}$$

where x_i is the value of the *i-th* feature in an individual example, whereas X_i is the entire sample. In 2015, Ioffe and Szegedy [108] presented the idea of *batch normalization*, also called *batchnorm* for short, devised to scale features in intermediate layers. This new kind of layers allowed to train significantly deeper models without the need for careful parameter initialization. Basically, in case of batch normalization, each

single feature map in a mini-batch of examples is to be scaled with Z-score normalization. Suppose T is a tensor corresponding to a mini-batch of 2D images, then T has the shape (B, C, H, W), in which B - is a number of images in the mini-batch, C - is a number of channels, or feature maps, whereas H and W determine the spacial size of the tensor T. Therefore, batchnorm layers first normalize the input tensor T as follows:

$$T'_i = \frac{T_i - \mathbb{E}[T_i]}{\sqrt{Var(T_i)}}\,, \tag{2.38}$$

where T_i denotes the *i-th* feature map of T. Then, batchnorm layers learn a pair of parameters γ_i and β_i to scale and shift each normalized channel:

$$\hat{T}_i = \gamma_i T'_i + \beta_i\,, \quad i = 1, \ldots, C\,. \tag{2.39}$$

Note that the output of batchnorm layers depends on the examples in the mini-batch, since they affect the mean and standard deviation. Therefore, batchnorm layers track moving averages of these values in the course of training and apply the computed statistics during inference to obtain deterministic predictions. Based on the concept of batch normalization, other normalization layers [9, 215, 231] have been devised for different learning tasks. Moreover, normalization layers have become a de facto standard for most modern architectures, wherein they are usually placed between each convolutional layer and nonlinearity.

In case of models with tens of convolutional (or feedforward) layers, normalization layers allow to reduce the problem of vanishing / exploding gradients that hamper convergence of gradient descent methods. However, training of deeper models usually requires to rely on *skip connections* and the principle of *residual learning*. He et al. [89] introduced a *residual unit* as a following transformation:

$$\begin{aligned} y^{(i)} &= h(x^{(i)}) + F(x^{(i)})\,, \\ x^{(i+1)} &= f(y^{(i)})\,, \end{aligned} \tag{2.40}$$

where $x^{(i)}$ is the input tensor to the *i-th* residual unit, F is a *residual function*, and h is known as a *skip connection*. Functions h and f are often set as an identity mapping, which leads to the residual unit with a *shortcut connection*, defined as

$$x^{(i+1)} = x^{(i)} + F(x^{(i)})\,, \tag{2.41}$$

which means that the input tensor $x^{(i)}$ could be directly propagated through the residual unit in both forward and backward passes. As a result, it has become possible to train extremely deep models with various types of residual units and skip connections (e.g., a 1001-layer network developed by He et al. [91]) using optimization algorithms based on gradient descent. Huang et al. [103] proposed to use a different connectivity patter to facilitate the information flow between layers by replacing summation with concatenation. Consequently, the unit with a *dense connection* has the output

$$x^{(i+1)} = \left[x^{(i)},\, F(x^{(i)})\right], \tag{2.42}$$

where $\left[x^{(i)},\, F(x^{(i)})\right]$ refers to the concatenation of the feature maps $x^{(i)}$ and $F(x^{(i)})$ along the channel dimension.

2.5 Architecture Design

Despite a myriad of existing CNN architectures applied to visual recognition tasks, the vast majority of them are built on the same underlying principles. A typical CNN has a *feature extractor*, also called an *encoder*, that converts an input data into a new, low-dimensional representation using a number of convolutional and pooling layers. Each convolutional layer is composed of multiple filters aiming to extract diverse visual patterns and generate corresponding feature maps. The first layers detect elementary, low-level features in the input data such as oriented edges, end-points, corners, etc. These features are then combined by the subsequent convolutional layers in order to extract more complex, high-level features with a larger *receptive field*[2]. In theory, the size of the receptive field grows linearly with the model depth, as each convolutional layer increases the receptive field by its kernel size. On the other hand, pooling layers enlarge the receptive field multiplicatively by reducing the feature map resolution [150], and help to learn representations that are invariant to small translations and distortions of the input [130]. Since pooling layers eliminate a part of spatial information encoded in feature maps, the width of subsequent convolutional layers is typically increased to compensate for the information loss. Thus, the encoder is a stack of alternating convolutional layers (followed by nonlinearity, e.g., ReLU)

[2]Since each filter in a convolutional layer is applied locally, a unit of the resulting feature map is only affected by a small neighborhood in the layer input. Likewise, in an entire CNN, the value of a unit in a certain feature map depends only on some region in the model input. This region is called a *receptive field* for that unit.

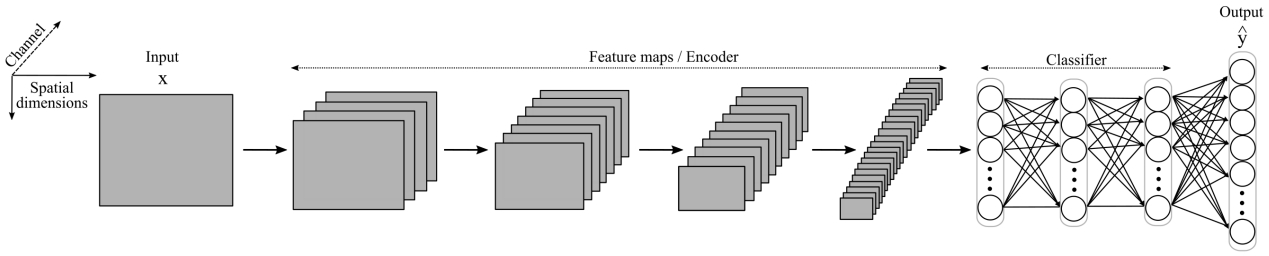


Figure 2.7: General encoder-classifier structure used in modern CNNs for image classification tasks.

and pooling layers, which returns a compact representation of the input.

In image classification tasks, the encoder is followed by a *classifier*, usually comprised of one or more fully connected layers. Given the compact representation of the input data produced by the encoder, the classifier aims to generate the correct probability distribution over the target classes using the softmax function (see Figure 2.7). This design principle was introduced in LeCun et al. [129] and LeCun et al. [130] to build LeNet, a pioneering network, successfully applied to handwritten digit recognition. Variants of this basic design have become prevalent in different vision recognition tasks and, most notably, in the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) [188], wherein, starting with AlexNet in 2012 [127], state-of-the-art results have been obtained exclusively by CNNs with the encoder-classifier structure (for example, see [91, 99, 103, 209]). In addition, decomposing CNNs into the encoder and classifier components leads to the appealing idea of *transfer learning*, an approach often used in practice [233]. Hypothetically, the whole encoder or its part, trained for some learning task, can be successfully reused in another related problems. Therefore, it is often advantageous to use encoders pretrained, for example, on the ImageNet data, instead of fitting the whole model from scratch.

Another important class of visual recognition tasks is *image segmentation*. In case of *semantic segmentation*, it is required to label all units (*pixels* in 2D images and *voxels* in 3D images) of the input with a certain category. In fact, this task can be thought of as a pixel- or voxel-wise classification problem, often referred to as *dense prediction*. The prediction is typically a *segmentation mask* of the same spatial size as the input, in which the probability distribution over all categories is provided for each unit. The term "semantic" emphasizes the fact that different objects, or instances, of the same category have to have the same label. On the other hand, in *instance segmentation*, unique labels must be given to each instance, even if they belong to the same category. For example, in radiotherapy, it is often necessary to label all units

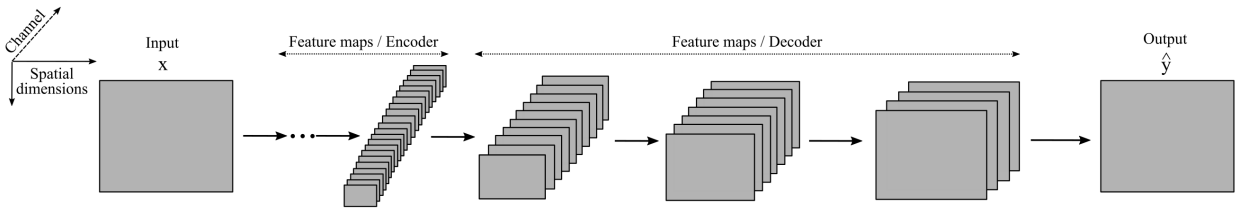


Figure 2.8: General encoder-decoder structure used in modern CNNs for image segmentation tasks. The encoder is shown in Figure 2.7 and omitted here for simplicity.

of a patient scan in order to locate all malignant lesions. If this task is formulated in the context of semantic segmentation, all lesions will have to be marked with the same label. Otherwise, if the individual label is required for each lesion, it will lead to the instance segmentation problem.

One of the first approaches for semantic segmentation with the use of CNNs appeared in Ciresan et al. [35] for segmentation of neuronal membranes in electron microscopy images (see details in Section 3). The proposed CNN predicted labels for a single pixel from raw image values in a square window centered on it. To get the segmentation mask for the whole image, the network required multiple forward passes in a sliding-window setup, resulting in significant computational overhead. Long et al. [144] introduced *fully convolutional networks* (FCNs), that did not have fully connected layers, could process inputs of an arbitrary size, and produced correspondingly-sized outputs with more efficient inference. The FCN included a pretrained encoder (based on AlexNet [127], VGG [198] and GoogLeNet [208]) to extract feature maps of different hierarchy and spatial size. Transposed convolutions and bilinear interpolation were applied to feature maps of different resolution to subsequently combine fine and coarse features, and eventually restore the input image size. SegNet was introduced by Badrinarayanan et al. [10] and incorporated an encoder similar to FCN and a *decoder*, a part of the network aiming to restore the input resolution, that led to the encoder-decoder structure (see Figure 2.8). Since the encoder provides high-level features with low spatial resolution, the main purpose of the decoder is to gradually upsample these features in order to generate the corresponding segmentation mask. Similarly, Ronneberger et al. [183] designed U-Net, arguably the most popular network for medical image segmentation, that includes skip connections to copy some feature maps from the encoder to the decoder (see details in Section 3).

Another class of methods, working especially well in instance segmentation tasks, is based on the idea of combining semantic segmentation with object detection [92,

179, 181]. First, these methods detect each individual object identifying a bounding box around it. Then, this bounding box is used to generate the segmentation mask corresponding to the object. Even though this approach leads to state-of-the-art results in many different applications with 2D images, it is barely employed in case of 3D medical scans. The main reason of it is that the detection task requires a large number of annotated objects available for training, which is quite rare in medical datasets.

2.6 Universal Approximation Theorems

The representational power, or the capacity, of feedforward neural networks has been widely studied in different fields. Most of these results are directly related to a number of *universal approximation theorems*, presented by Cybenko [44], Hornik et al. [97] and Hornik [98]. For example, Theorem 2 from [44] can be formulated as follows:

Theorem 2.6.1 *If F^* is a continuous function defined on the p-dimensional unit cube I_p, i.e., $I_p = [0,1]^p$. Then, for any given $\varepsilon > 0$, there exists*

$$F(\mathbf{x}) = \sum_{j=1}^{N} \alpha_j \sigma(\mathbf{w}_j^\top \mathbf{x}) \,, \tag{2.43}$$

such that for all $\mathbf{x} \in I_p$

$$|F(\mathbf{x}) - F^*(\mathbf{x})| < \varepsilon \,. \tag{2.44}$$

The transformation $\mathbf{w}_j^\top \mathbf{x}$ corresponds to an affine transformation of the input $\mathbf{x}$, and σ is a "squashing function", i.e., any continuous function of the form

$$\sigma(z) = \begin{cases} 1 & \text{if } \; z \to +\infty \\ 0 & \text{if } \; z \to -\infty \end{cases} \tag{2.45}$$

In other words, Theorem 2.6.1 states that any continuous function can be approximated by a *shallow feedforward neural network*, i.e., a network with one hidden layer (see Figure 2.9), and with an arbitrary squashing function, e.g, the sigmoid activation. This type of networks is therefore said to be a universal approximator. Leshno et al. [133] and Pinkus [174] extended results of the universal approximation theorem to a wider class of activation functions, including rectified linear units.

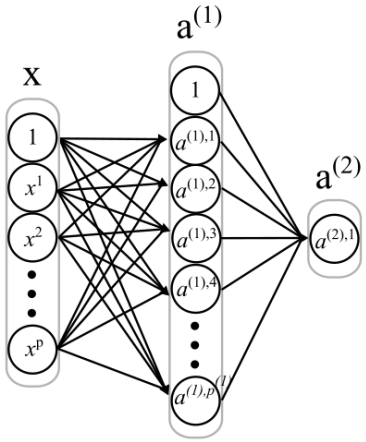


Figure 2.9: Shallow feedforward neural network.

The presented theorem says that there exists a network large enough to achieve any desirable accuracy, however, it does not say anything about the size of the hidden layer. In fact, this single layer may be infeasibly large so that the network will fail to learn and generalize correctly. In many circumstances, using deeper models can significantly reduce the number of neurons required to approximate the desired function [137]. It was demonstrated that some functions require exponentially many neurons in a shallow network to achieve the same approximation accuracy as a deep network with only a polynomial or linear number of neurons [40, 60]. However, if the input length is p, even an infinitely deep feedforward network with ReLU nonlinearities must have at least $p + 1$ neurons in each hidden layer to be the universal approximator [72, 147]. Lin and Jegelka [137] showed that feedforward networks with shortcut connections and just one neuron representing the residual function F (see Equation 2.41) is enough to provide universal approximation as the depth goes to infinity. This result implies that, compared to fully connected layers, the identity mapping in residual layers allows to improve the representational power of deep networks. However, both deep and wide networks require a large number of training examples. In addition, nonlinear activation functions, used in hidden layers, make most of loss functions to become non-convex. Gradient descent methods applied to non-convex loss functions have no global convergence guarantees and are sensitive to the values of the initial parameters. Although the universal approximation theorem states that for any continuous function there exists its approximation with a shallow network, it does not guarantee that the training algorithm will be able to correctly learn that network from the training set. This situation can be described in terms of the bias-variance tradeoff (see Section 1.5). The universal approximation theo-

rem guarantees that the approximation error of shallow networks is zero. However, the estimation error might be extremely large, making these models inapplicable in practice for small training sets.

Similar theoretical results on the approximation capacity of CNNs (without fully connected layers) were provided recently. Zhou [240], for example, proved the universality of CNNs to approximate any continuous function to an arbitrary accuracy, if the model depth is large enough. These results verified the efficiency of deep CNNs in dealing with large dimensional data.

Chapter 3

Overview of CNNs for Medical Image Segmentation

From Classification to Segmentation

To the best of my knowledge, the first convolutional neural network (CNN) applied to a medical visual recognition task dated back to 1995, when Lo et al. [142] used it for lung nodule classification in CT images. Their model consisted of two convolutional layers and operated on extracted image patches with a size of 16×16 pixels. Despite its simplicity, the model achieved high accuracy (AUC = 0.83) on 207 testing patches and set an important precedent for the future use of CNNs in medical image segmentation.

It is generally accepted that CNNs owe much of their success to AlexNet [127] that won the ImageNet LSVRC-2012 competition by a large margin. However, the other network, called DanNet [36], won four other image recognition challenges in a row prior to AlexNet. In fact, DanNet was the first CNN that surpassed human-level performance (twice as good as humans) in a vision challenge [37]. This network also became the first CNN to win a contest on object detection in large images ($2048 \times 2048 \times 3$ voxels) and a medical imaging contest [38, 39]. In addition, DanNet was the first CNN that outperformed classical methods for image segmentation in a public challenge. More specifically, in the ISBI 2012 EM Segmentation Challenge, DanNet was applied to segment biological neuron membranes in stacks of electron microscopy (EM) images [35]. Since DanNet is an encoder-classifier network (see Section 2.5), it used a square image patch as input to classify just a single central pixel in the entire patch. After training, to segment a whole test image, the model was applied to all image pixels in a sliding-window fashion (see Figure 3.1) that made this method computationally expensive. It is worth noting that DanNet has a striking

similarity to AlexNet in terms of architecture design, and it is also one of the first CNNs that relies on GPU-based training.

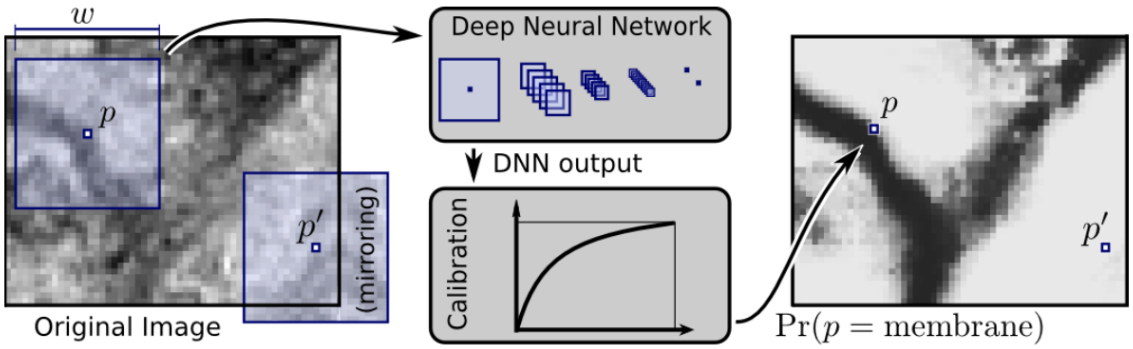


Figure 3.1: DanNet for EM image segmentation. An image patche with a size of $w \times w$ pixels is used to compute the probability of a central pixel being a membrane. Source: Ciresan et al. [35].

U-Net as a Starting Point for Architecture Design

As stated earlier in Section 2.5, encoder-decoder networks (e.g., FCN [144] and SegNet [10]) overcome drawbacks with the sliding-window approach in segmentation tasks by employing a decoder to generate the prediction for the entire input with a single forward pass. U-net [183] is an instance of this type of networks that includes skip connections to transfer some feature maps from the encoder to the decoder. In case of medical image segmentation, the vast majority of models have been built on the basis of U-Net.

In the U-Net architecture, at each resolution stage (i.e, before downsampling / upsamling), two 3×3 convolutional layers followed by ReLU are applied to generate feature maps. After downsampling, usually performed by a 2×2 max pooling, the number of feature channels is doubled. In the decoder, upsampling is done by 2×2 transposed convolutions that halve the number of feature channels. Then, the upsampled maps are concatenated with the corresponding maps from the encoder, and two 2×2 convolutional layers with ReLU are applied to combine these maps. The final layer is a 1×1 convolution that generates the proper number of output classes. The last activation function returns the probability distribution over the classes for each pixel. The original U-Net introduced by Ronneberger et al. [183] is shown in Figure 3.2. By replacing all 2D layers with their 3D counterparts, one can get the 3D U-Net architecture [34], widely used in medical image analysis. Today, it is common practice to apply normalization (e.g., batch normalization) after each convolutional

layer and before nonlinearity to facilitate training. The combination of these three transformations is defined as a *convolutional block*.

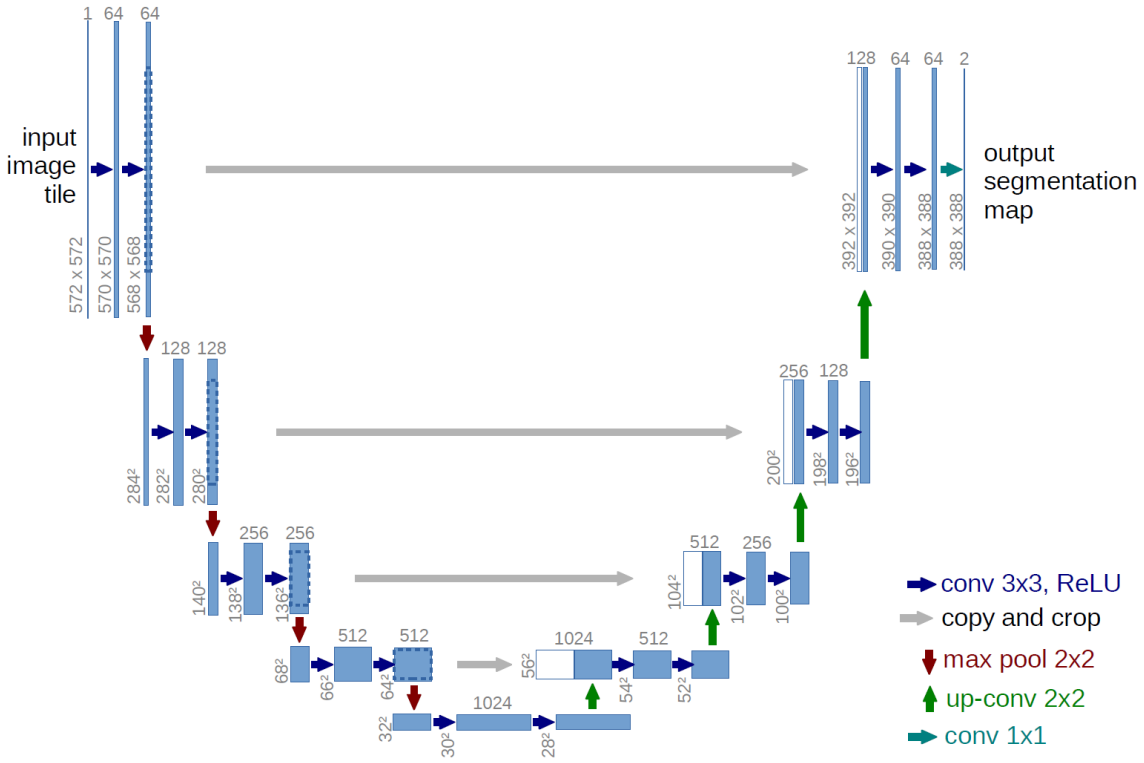


Figure 3.2: Original 2D U-Net with 64 filters in the first convolutional layer. Source: Ronneberger et al. [183].

Towards New Blocks and Layers

Many architectures have been derived from U-Net by relying on *alternative convolutional blocks*. For example, Milletari et al. [158] introduced V-Net for prostate segmentation in transversal T2-weighted MR images. This network used residual blocks [87] in both encoder and decoder, and applied strided convolutions instead of max pooling for downsampling. In addition, this paper presented the Soft Dice Loss, a differentiable surrogate for the Dice similarity coefficient (DSC) (defined in Section 6.2.3), which is commonly used in modern applications. Likewise, Dolz et al. [51] used dense blocks [101] as the basic unit of their model and additional skip connections between encoding and decoding paths to address two different brain tissue segmentation tasks in MRI. Alom et al. [4] proposed a recurrent U-Net (RU-Net) model and a recurrent residual U-Net (R2U-Net) model that outperformed U-Net in three different tasks, namely blood vessel segmentation in retinal images, skin cancer

segmentation and lung lesion segmentation. In these models, all convolutional blocks were replaced with novel recurrent residual convolutional units that relied on skip connections and the general ideas of recurrent neural networks. Jin et al. [118] proposed a 3D hybrid residual attention-aware segmentation method (RA-UNet) based on U-net with integrated attention residual modules. This model was successfully applied to the task of liver tumor segmentation in CT images. Jha et al. [115] designed an architecture with diverse types of residual layers for colonoscopic image segmentation. Their model outperformed state-of-the-art methods and the original U-Net on two public datasets. Wang et al. [222] described another network based on 3D U-Net with recursive residual blocks and pyramid pooling for segmenting three brain sub-regions, namely white matter, gray matter and cerebrospinal fluid, in T1-weighted MRI. The recursive residual blocks contained multiple skip connections to ease training, while pyramid pooling was applied to generate feature maps at different resolution stages for obtaining both local and global image features. In addition, the deep supervision mechanism was incorporated into the network. On three public datasets, the proposed model generated accurate predictions for all brain sub-regions with the average DSC of 0.91, 0.90 and 0.85 on the corresponding datasets.

Attention to Details

Another common source of modification is to supply U-Net with *various attention mechanisms* [11]. Oktay et al. [165], for instance, presented an extension to the standard U-Net model for gastric cancer segmentation in CT images by incorporating new attention gate (AG) modules in the decoder. These modules aimed to suppress irrelevant regions in the input image while highlighting salient features useful for a specific task. In a similar manner, Roy et al. [184] proposed concurrent spatial and channel squeeze & excitation (scSE) modules to adaptively recalibrate feature maps during training. Integrating these lightweight modules into fully convolutional neural networks provided consistent improvements of performance across different architectures, which was demonstrated in the task of segmenting MRI T1 brain scans into 27 cortical and sub-cortical structures as well as for organs delineation in whole-body contrast-enhanced CT scans. An alternative attention mechanism, convolutional block attention module (CBAM), was successfully applied to nasopharyngeal carcinoma segmentation in multisequence MRI [228]. Zhou et al. [242] used scSE modules similar to Hu et al. [99] and Roy et al. [184] in order to add the attention mechanism to various parts of the U-Net architecture, including skip connections. These modules re-weighted feature maps channel- and space-wise to obtain more

informative feature representations. In addition, residual blocks with dilated convolutions were used to construct both encoder and decoder with the larger receptive field. The model was trained to delineate three target classes in CT scans of patients with COVID-19.

New Architectures

The broad range of models are in fact *deviations from the traditional U-shaped architecture* shown in Figure 3.2. For instance, Myronenko [161] supplemented the encoder-decoder network comprised of full pre-activation residual blocks [88] with an additional variational auto-encoder branch that aimed at reconstructing the input image. It imposed additional constraints on the model weights and therefore served as a form of model regularization. This architecture demonstrated the best results in the Brain Tumor Segmentation (BraTS) Challenge 2018 [12]. [245] addressed the task of chronic stroke lesion segmentation in 3D MRI T1 scans by doubling the encoder path. Each branch of the encoder extracted features of the specific dimension (2D or 3D) that were subsequently combined and transferred to the decoder. It achieved better performance compared to 2D counterparts while significantly reducing computational overhead in comparison to pure 3D networks. Zhou et al. [247] re-designed skip connections in U-Net by including a series of nested dense blocks that increased semantic similarity between the encoder and decoder feature maps. The model was superior to different variants of U-Net for nodule segmentation in the low-dose chest CT scans, nuclei segmentation in the microscopy images, liver segmentation in abdominal CT scans, and polyp segmentation in colonoscopy videos. Li et al. [136] presented an architecture consisting of multiple iterations of U-Net with additional skip connections. Each iteration was designed to refine predictions obtained in previous steps. The model achieved state-of-the-art performance in retinal vessel segmentation on three commonly used datasets. In the context of the automatic liver segmentation in CT, Dou et al. [52] provided the model with multiple auxiliary branches aiming to generate predictions with different resolutions. Each auxiliary output was first upsampled to the proper output size with the use of transposed convolutions. Then, the weighted average of the generated outputs was used to produce the final prediction. Often, this procedure is referred to as *deep supervision* [131]. Finally, a fully connected conditional random field (CRF) method was applied as a post-processing step to refine the output. On the public dataset, this approach outperformed all other methods while demonstrating the best running time. Another variant of 3D U-Net with deep supervision was described in Kayalibay et al. [120]. In this network, convolutional

blocks were replaced with residual blocks, and strided convolutions were employed for downsampling in place of max pooling. Each pair of feature maps at adjacent stages in the decoder was subsequently combined by element-wise summation (that corresponds to deep supervision) to compute the prediction. Hence, the output was directly affected by the features from all stages that helped to speed up convergence during training. This model was used to segment four classes of human phalanges in hand MRI.

Multibranch Structures

In the BraTS 2019, Jiang et al. [117] merged two U-Net architectures into a two-stage cascaded model to address the task of glioma sub-region segmentation in multisequence MRI. During the first stage, the first U-Net was trained on the original input to produce coarse, preliminary predictions. Then, the original input and corresponding predictions were merged and subsequently used as input by the second U-Net to refine the preliminary results. The final prediction was obtained after applying post-processing to the output of the second U-Net. Although the whole model was relatively cumbersome and could not produce end-to-end predictions, this method won first prize in the BraTS 2019. It is worth noting that this approach is a special case of two-level stacking introduced in Section 1.8. Isensee and Maier-Hein [111] applied three variants of U-Net in the Kidney Tumor Segmentation (KiTS) Challenge 2019 [94]. In addition to the standard 3D U-Net architecture, they designed two additional models with two types of residual blocks [87, 89] used solely in the encoder. Despite the large number of publications claiming substantial advantages of residual layers over basic convolutional blocks, all three models obtained results without significant differences.

Instead of using a fixed-size patch as input, Kamnitsas et al. [119] presented DeepMedic, a dual pathway architecture, to process the input images at *multiple scales* simultaneously. In this model, the original patch and its downsampled counterpart were passed through individual pathways to extract both local and larger contextual features, which were subsequently fused to generate the output. Similarly, Srivastava et al. [202] designed a dual-scale dense fusion block to compute both high- and low-resolution feature representations. U-Net supplemented with these blocks achieved state-of-the-art results on four public medical datasets. For meningioma segmentation in Gd-enhanced T1-weighted MRI, Bouget et al. [23] designed U-Net with multi-scale input, attention and deep supervision. At each resolution stage, the downsampled input obtained with the use of average pooling was concatenated

to the corresponding feature maps in the encoder. Likewise, downsampled ground-truth masks were used to evaluate the intermediate loss at each stage in the decoder to apply deep supervision. The total value of the loss function was the sum of all intermediate losses. Two different types of attention mechanisms were applied to skip connections to highlight relevant feature maps, before transferring them to the decoder. Using 5-fold cross-validation (three folds were used for training, one for validation, and one for testing) on a dataset with about 600 patients, the best model configuration obtained the average DSC of 0.82 across all test folds.

Also, a variety of fusion techniques have been proposed in multimodal image segmentation to efficiently combine and/or process diverse image modalities [104, 243, 244].

Chapter 4

Cervical Cancer Segmentation in PET

Reference

Convolutional neural networks for PET functional volume fully automatic segmentation: development and validation in a multicenter setting. Andrei Iantsen, Marta Ferreira, Francois Lucia, Vincent Jaouen, Caroline Reinhold, Pietro Bonaffini, Joanne Alfieri, Ramon Rovira, Ingrid Masson, Philippe Robin, Augustin Mervoyer, Caroline Rousseau, Frédéric Kridelka, Marjolein Decuypere, Pierre Lovinfosse, Olivier Pradier, Roland Hustinx, Ulrike Schick, Dimitris Visvikis, and Mathieu Hatt. *European Journal of Nuclear Medicine and Molecular Imaging* (2021).

Abstract

Purpose: In this work, we addressed fully automatic determination of tumor functional uptake from positron emission tomography (PET) images without relying on other image modalities or additional prior constraints, in the context of multicenter images with heterogeneous characteristics. **Methods:** In cervical cancer, an additional challenge is the location of the tumor uptake near or even stuck to the bladder. PET datasets of 232 patients from five institutions were exploited. To avoid unreliable manual

delineations, the ground-truth was generated with a semi-automated approach: a volume containing the tumor and excluding the bladder was first manually determined, then a well-validated, semi-automated approach relying on the Fuzzy locally Adaptive Bayesian (FLAB) algorithm was applied to generate the ground-truth. Our model built on the U-Net architecture incorporates residual blocks with concurrent spatial squeeze and excitation modules, as well as learnable non-linear downsampling and upsampling blocks. Experiments relied on cross-validation (four institutions for training and validation, and the fifth for testing). **Results:** The model achieved good Dice similarity coefficient (DSC) with little variability across institutions (0.80±0.03), with higher recall (0.90±0.05) than precision (0.75±0.05) and improved results over the standard U-Net (DSC 0.77±0.05, recall 0.87±0.02, precision 0.74±0.08). Both vastly outperformed a fixed threshold at 40% of SUVmax (DSC 0.33±0.15, recall 0.52±0.17, precision 0.30±0.16). In all cases, the model could determine the tumor uptake without including the bladder. Neither shape priors nor anatomical information was required to achieve efficient training. **Conclusion:** The proposed method could facilitate the deployment of a fully automated radiomics pipeline in such a challenging multicenter context.

4.1 Introduction

Combined positron emission tomography / computed tomography (PET/CT) imaging is widely used in clinical practice to provide functional information on organs and tissues, as well as disease abnormalities. Static PET images provide semi-quantitative information regarding the distribution of a radiotracer uptake. In oncology, Fluorodeoxyglucose (FDG) PET imaging is routinely relied upon for diagnosis, staging, treatment planning and therapy follow-up [82].

In clinical applications, nuclear medicine physicians carry out qualitative assessments of PET/CT images, which is typically sufficient for detecting and anatomically locating lesions. For radiotherapy treatment planning, radiation oncologists manually draw boundaries on fused PET/CT images to determine the gross target volume (GTV) of a tumor, in order to subsequently deliver a specific dose to the target. The boundary of the target should be defined as accurately as possible to maximize the

coverage of the target and minimize the dose delivered to surrounding healthy tissues and nearby organs-at-risk (OAR).

More quantitative assessment of FDG uptake in PET images can also be performed. For instance, radiomics analyses [85] aim at extracting clinically relevant measurements through the calculation of numerous image derived features features (e.g., intensity, shape and textural). Such measures can subsequently be used to build models predictive of outcome or for assessing changes in tumors before, during, and after treatment in order to better evaluate response to therapy [47, 148, 151]. It has been shown in all image modalities including PET that the choice of the segmentation method in this step of the radiomics workflow can significantly affect the extracted features [84, 123, 173, 182, 236]. In addition, it is recognized that in the absence of fully automated segmentation, this step is a crucial bottleneck and time-consuming step of any radiomics study, preventing such a process to be expanded to very large datasets [85]. There is therefore a need for a delineation method that is not only accurate and robust, but also fully automated as well.

There are several challenges pertaining to PET image segmentation [82]. First, PET images suffer from limited spatial resolution (4-5 mm), comparatively to CT (below 1 mm) due to partial volume effects (PVE) that make boundaries between adjacent functional regions blurred and result in under-estimated activity in small objects of interest. Second, signal-to-noise ratio in PET images is inherently low and affected by a vast array of factors, such as scanner sensitivity, temporal resolution, acquisition mode, scan time, quantity and distribution of tracer, applied corrections (e.g., scatter, attenuation, randoms) and reconstruction algorithm type (e.g., resolution recovery, time of flight) and parameters (e.g., number of iterations). All these issues make things challenging in a multicenter context, i.e., when analyzing PET images acquired using different systems, acquisition protocols and reconstruction settings. Third, the wide variability in shapes and heterogeneity of lesion uptakes might reduce the generalization of segmentation methods to only some specific cases.

An important aspect in medical image segmentation is that the true boundary of the object of interest (ground-truth) is impossible to determine without a complete histopathological analysis of an excised tumor, which can typically be performed only in a small number of cases. In PET, even with a very robust protocol, this approach can only provide approximate co-registration between the histopathology slides and the corresponding 3D PET slices [82]. One way to overcome this is the use of a consensus of several manual segmentations by experts as a surrogate of truth [82]. Unfortunately, manual segmentation is typically a labor intensive, time consuming

task with low reproducibility, due to the high intra- and interobserver variability [79].

There have been a number of algorithms proposed for PET image segmentation, accounting at different degrees for some of the limitations referred to above [82]. For example, thresholding-based methods, the most simple image segmentation techniques, work on the assumption that different tissue types have specific uptake ranges; therefore, segmentation can be done by comparing individual voxel intensities with a set of thresholds. More advanced methods aim at exploiting statistical differences between uptake regions and surrounding tissues. These include different clustering and classification methods trained on a set of features extracted from PET images, as well as atlas based [154, 180] and generative models such as Gaussian mixture models (GMM) [8] and Fuzzy Locally Adaptive Bayesian (FLAB) model [77]. Numerous other common image segmentation algorithms have been evaluated for this task using PET uptake only [82]. For the vast majority of these published methods, it is usually assumed that the tumor has been previously isolated in a volume of interest (VOI), i.e., the input to the algorithm is not the entire PET image but a sub-volume containing the object of interest, that is usually manually determined after visually detecting the tumor uptake in the whole image. It should be emphasized on that numerous approaches tried to improve PET segmentation by considering both PET and CT modalities together, assuming an (almost) perfect correspondence between tumor functional uptake and tumor anatomical boundaries as determined on CT images using co-segmentation approaches exploiting co-registered PET and CT images [59, 74, 135]. This assumption may not be true as radiotracer uptake and anatomical boundaries can be uncorrelated. This also makes the method sensitive to registration issues in PET imaging, especially in body regions affected by motion [82].

Convolutional neural networks (CNNs) have been successfully applied to different medical imaging tasks [141], such as reconstruction [132, 177], denoising [67, 178], segmentaton [70, 158] and classification [41]. Most segmentation studies rely on U-Net [183], that is arguably the most popular network for semantic segmentation, and focus on anatomical modalities such as magnetic resonance imaging (MRI) [51, 158] and CT [165, 184]. The limited number of papers dedicated to PET segmentation usually assume a correspondence between functional and anatomical regions in combined PET/CT or PET/MRI imaging [71, 100, 237–239]. The ground-truth is usually obtained through manual delineation performed on multimodal images (e.g., training a network to reproduce delineations performed by radiation oncologist that perform this manually on fused PET/CT images). Guo et al. [70] included PET imaging within a CNN based multimodal image segmentation framework using PET, MR (T1

Table 4.1: Summary of patients, including the different characteristics of the scanners, and associated reconstruction methods and parameters.

Institution	Numb. of patients	Scanner	Voxel sizes (mm^3)	Reconstr. methods	Time per bed position (s)	FDG total dose (MBq)
University Hospital of Brest, France	69	Siemens Biograph	$4.073 \times 4.073 \times 2.027$	PSF+TOF	120 ± 17	253 ± 80
Integrated Centre for Oncology (ICO), France	18 5	Siemens Biograph GE Discovery STE	$4.073 \times 4.073 \times 2.027$ $4.688 \times 4.688 \times 3.27$	PSF+TOF 3D IR	202 ± 32	210 ± 56
McGill University Health center, Canada	7 19	GE Discovery 710 GE Discovery ST	$3.646 \times 3.646 \times 3.27$ $5.469 \times 5.469 \times 3.27$	VPFXS OSEM	212 ± 30	398 ± 81
Hospital of the Holy Cross and Saint Paul, Spain	24	Philips Gemini TF	$4 \times 4 \times 4$	BLOB-OS-TF	109 ± 21	228 ± 50
University Hospital of Liège, Belgium	90	Philips Gemini TF	$4 \times 4 \times 4$	BLOB-OS-TF	73 ± 16	260 ± 32

and T2) and CT images of a publicly available soft tissue sarcoma dataset of 50 patients. Gross tumor volumes were manually annotated in all four imaging modalities. Different fusion networks were used for feature-, classifier- and decision-level fusion, demonstrating improved performance at a feature level fusion [70].

Considerably less attention has been dedicated to processing PET images as a standalone modality. Moreover, the majority of studies have used only datasets with small cohorts of patients from one or two centers and manual delineation as a surrogate of truth. Under these circumstances, some previously published results might be less generalizable due to high heterogeneity of PET images caused, for instance, by scanner type, reconstruction algorithm and applied post-processing that vary across centers. Huang et al. [100] applied U-Net with minor modifications for head and neck cancer gross tumor volume segmentation on PET/CT images. Results were obtained for a dataset of 22 patients using manual segmentation as a surrogate of truth. Blanc-Durand et al. [22] evaluated U-Net for glioma segmentation on PET images with the fluoroethyl-tyrosine (FET) tracer. Their dataset contained only 37 patients with manually segmented lesions. Leung et al. [134] used a modified U-Net trained on simulated PET images and fine-tuned using a clinical dataset of 160 patients with manual delineations for lung cancer segmentation. In cervical cancer, Chen et al. [32] proposed to combine a 2D CNN and a post-processing step that relies on prior anatomical information on the tumor roundness and its position relative

to the bladder. The choice of the 2D network was dictated by the limited size of the available dataset that contained 1176 slices from 50 patients, and the surrogate of truth was also obtained through manual delineation. Within the scope of the recent MICCAI challenge on automatic PET tumor functional volume delineation, the CNN-based method reached the highest score compared to twelve other approaches, amongst them some of the current state of the art [83].

In this paper, we focused our experiments on cervical cancer. Approximately 570 000 new cases of cervical cancer and 311 000 deaths from the disease occurred in 2018 and this type of cancer was the fourth most common cancer in women worldwide [7]. Recently, predictive models relying on textural features from tumor volumes in PET images were able to identify the subset of patients that will suffer from recurrence after treatment with clinically-relevant accuracy [148, 182]. This obviously requires accurate delineation of the tumor volume in the PET images and this is achieved without the use of the associated CT image. Due to the anatomical proximity between the cervix and the bladder that generally has similar FDG uptake in PET scans, conventional techniques (e.g., thresholding, region- and boundary-based methods) provide poor results if applied to the whole image without additional prior knowledge or constraints, which is why a VOI excluding physiological uptake usually needs to be provided as input to the method. For instance, the use of FLAB, as described in the radiomic study above [148], requires an expert to first manually define a VOI containing the tumor but excluding the bladder, in which FLAB is then applied to delineate the tumor. However, this step can be quite labor intensive and time-consuming, especially when the tumor uptake and bladder are very close to each other, hindering the potential clinical translation of these segmentation tools, and in turn the use of the predictive radiomics based models. A fully automated segmentation step, without the manual determination of the VOI, is therefore highly desirable in that context.

The purpose of our study was thus to propose a U-Net based model for the fully automated delineation (i.e., without the need for visual detection of their location and manual determination of a VOI) of 3D functional primary tumor volumes in PET images only, especially in the specific context of cervical cancer where the pathological uptake of interest is located close to a physiological one that should not be included (here, the bladder). A secondary objective was to train the network on a reliable ground-truth obtained through accurate and robust PET semi-automated segmentation instead of manual delineation. A final objective was to train and evaluate the performance of our model under standard clinical imaging conditions, considering a

multicenter patient cohort without any prior standardization in the data acquisition or image reconstruction processes.

4.2 Materials and Methods

4.2.1 PET Images and FLAB-derived Ground-Truth

Our first objective is to achieve fully automated determination of the functional uptake boundaries in PET images only, without relying on assumptions regarding its correspondence with anatomical boundaries and to avoid registration issues, which are important in the case of cervical cancer due to the elastic nature of organs in this body region. We decided to train and evaluate the proposed model exclusively on real, clinical images, contrary to recent recommendations by the task group 211 of the AAPM (American Association of Physicists in Medicine) dedicated on PET auto-segmentation, which advises to rely on a combination of simulated, phantom and clinical images [82]. Indeed, usually the number of clinical images available for training and validation is small, and the surrogate of truth is questionable when only manual delineations from experts are available. In such a context, results obtained on large datasets of simulated and/or phantom images can indeed increase the confidence in the results obtained in a smaller amount of clinical cases with less reliable surrogate of truth. However, in the present work, we exploited a large dataset of images that were processed by experts using a semi-automated approach (see section below detailing how the ground-truth was generated) for the purpose of radiomics-based outcome modeling studies. In addition, one objective of this work is to evaluate the ability of the proposed approach to deal with fully automated tumor uptake delineation when it is located close to a physiological one that should not be included. Simulated or phantom images corresponding to this specific context are currently not available in large amounts to train and evaluate a CNN-based approach.

We collected a dataset of 232 FDG PET images of patients from five institutions, all with histologically proven cervical cancer, with clinical stage IB1-IVB[1] (Figure A.1). All images contained the abdominopelvic cavity and were acquired for diagnostic and staging purposes before chemoradiotherapy followed by brachytherapy. Collected images considerably differed in acquisition protocols (scanning duration per bed position, injected radiopharmaceutical dose) and reconstruction (algorithms, use

[1] According to the staging system of the International Federation of Gynecology and Obstetrics (FIGO).

of time of flight information, resolution modeling, voxel and matrix sizes)(see Table 4.1). Data were corrected for randoms and scatter in all cases. All reconstructed images were corrected for attenuation using the associated low dose CT.

An objective of our work is to train the network using a reliable ground-truth excluding the bladder uptake. Segmentation of the tumor volumes to generate the ground-truth was performed on PET images using the FLAB algorithm [77] applied in a semi-automated manner: first, a VOI containing the tumor uptake and excluding the nearby bladder and other physiological uptakes was manually defined by the user. The FLAB algorithm was then run within that VOI to generate a segmentation mask that was reviewed by the user. If this mask was deemed unsatisfactory, the user had the option to re-run the algorithm after specifying different values of initialization parameters in order to obtain a more satisfying result. Finally, the results for all tumors were reviewed and in some cases (<5%) manually edited before being validated by two experts with more than 15 years of clinical practice. Given that FLAB in such a context has been demonstrated to provide accurate and reliable results in numerous studies [69, 84, 207], including for complex heterogeneous cases [78, 79] and over different scanner model and reconstruction algorithms, especially compared to manual delineation [80], we consider this ground-truth sufficiently reliable for the purpose of training and evaluating the proposed approach. Although FLAB was applied only within the manually determined VOI, we then registered the obtained segmentation mask onto the entire PET image used as input to the network for training and testing.

4.2.2 Network Architecture

The widely used 3D U-Net model [34] served as the basis for our network design. Although not the main objective of the present work, we nonetheless introduced three optimizations beyond the standard U-Net model:

1. Original U-Net consists of conventional convolutional blocks comprised of a $3 \times 3 \times 3$ convolution, a normalization layer (e.g. batch norm) and a ReLU activation function as a basic element of the network. We chose to rely upon a residual block with full pre-activation [88] supplemented by a concurrent spatial and channel squeeze & excitation (scSE) module (Figure 4.1, grey blocks).

2. An important part of the proposed architecture is the integration of squeeze and excitation (SE) blocks that aim at providing the option to compute weights for the feature maps, so the network can put more or less attention on some of them.

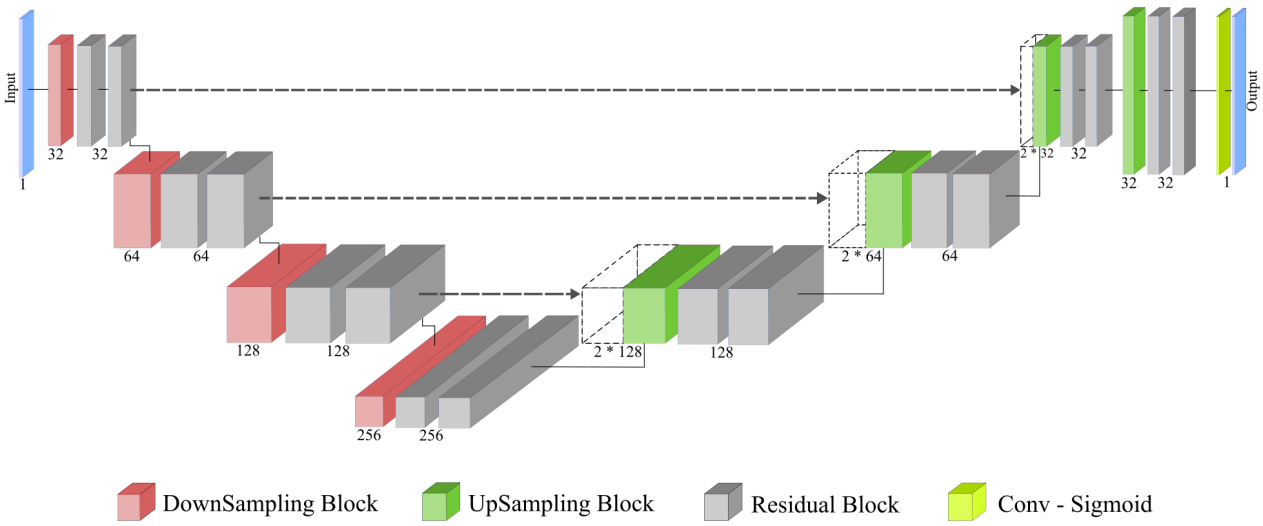


Figure 4.1: Proposed Encoder-Decoder Network with residual blocks. The number of output channels is depicted under blocks of each group.

We implemented SE blocks within the full pre-activation residual blocks, namely a specific modification called concurrent spatial SE (scSE), that has been shown to perform better for image segmentation tasks [184]. In order to include the scSE module in the residual block, we followed the same approach that was applied in SE-ResNet architectures [99]. Due to the high memory consumption working with 3D images, we switched from using batch norm layers to instance normalization (instance norm) that was shown to work better in a small-batch regime [230].

3. We replaced max pooling operations in the encoder of the network by learnable downsampling blocks (Figure 4.1, red blocks), which consist of one $3 \times 3 \times 3$ strided convolutional layer, the instance norm, the ReLU activation and the scSE module. Similarly, we implemented upsampling blocks in the decoder of the network using $3 \times 3 \times 3$ transposed convolutions instead (Figure 4.1, green blocks). To reduce memory consumption and increase the receptive field of the network, we implemented the first downsampling block with a kernel size of $7 \times 7 \times 7$ right after the input. The last convolutional layer followed by the sigmoid activation function to produce the model output was applied with a kernel size of $1 \times 1 \times 1$.

4.3 Experimental Settings

4.3.1 Data Preprocessing

The PET images exhibited a large variability of voxel sizes (see Table 4.1) that can adversely affect the model performance since CNNs cannot natively interpret spatial dimensions with different scales. Therefore, we first interpolated all PET images and corresponding segmentation masks to a common resolution of $4 \times 4 \times 2\ mm^3$ through the use of linear interpolation. A slice thickness of 2 mm was chosen to retain small image details that could be lost if interpolated at a larger voxel size. Linear interpolation was chosen after comparison with other techniques including Nearest Neighbour and B-spline, which led to decreased performance.

PET image intensities can exhibit a high variability in both within-image and between images. In order to reduce these variabilities and use the PET scans as the input for the CNNs, we applied Z-score normalization for each scan separately, with the mean and the standard deviation computed based only on voxels with non-zero intensities corresponding to the body region.

4.3.2 Data Augmentation

Due to the large variability in shapes, sizes and heterogeneity of tumor uptakes in PET images, data augmentation can play a useful role in model training. To aid the model learn features invariant to affine transformations that are realistic, we applied mirroring on the axial plane, rotations in random directions with the angle uniformly sampled from the range [5, 15] degrees along the random set of axes, and scaling with a random factor between 0.8 and 1.2. In order to increase the diversity in lesion shapes, we relied on elastic deformations. Gamma correction with γ sampled from the uniform distribution between 0.8 and 1.2, and contrast stretching between 0 and 0.8 - 1.2 of the original range of values were applied to adjust voxel intensity distributions. To improve model robustness, we also added Gaussian noise to training samples. The standard deviation of the noise was equal to 0.1 - 0.2 standard deviations of the training sample. All augmentation methods were applied independently during model training with a probability of 0.2.

4.3.3 Training Procedure

Due to the large size of PET images, we trained the model on randomly extracted patches of $128 \times 128 \times 64$ voxels with a batch size of 2. Since all PET images

corresponded to the abdominopelvic cavity with a number of axial slices ranging from 77 to 192, the chosen patch size was large enough to cover a significant part of the input PET image for all patients.

We trained the model for 400 epochs using Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.99$ for exponential decay rates for moment estimates. We applied a cosine annealing schedule [146], gradually reducing the learning rate from $lr_{max} = 10^{-4}$ to $lr_{min} = 10^{-6}$ for every 25 epochs and performing the adjustment at each epoch.

Considering the fact that the Dice similarity coefficient (DSC) is one of the most common metrics used for the evaluation in medical image segmentation, we trained the model with the Soft Dice Loss. Based on [158], in case of binary segmentation, the loss function for one training example can be written as

$$L(y, \hat{y}) = 1 - \frac{2\sum_i^{\mathcal{N}} y_i \hat{y}_i + 1}{\sum_i^{\mathcal{N}} y_i^2 + \sum_i^{\mathcal{N}} \hat{y}_i^2 + 1} \tag{4.1}$$

where $y_i \in \{0, 1\}$ - the binary label for the *i-th* voxel, $\hat{y}_i \in [0, 1]$ - predicted probability for the *i-th* voxel. Additionally we applied Laplacian smoothing by adding $+1$ to the numerator and the denominator in the loss function to avoid the zero division in cases when only one class is represented in the training example.

4.3.4 Multicenter Cross-Validation

Cross-validation is probably the simplest and most widely used method for estimating the expected prediction error of a model on an independent test sample [75]. Importantly, cross-validation is based on the assumption that data samples in the train and test folds are drawn from the same distribution. However, as already mentioned in Section 4.2.1, no standardization in the acquisition or reconstruction protocols were implemented across the five institutions in which the images were collected. In addition, different PET/ CT imaging devices with variable overall performance were used in these centers. Therefore, in order to obtain more reliable estimate of the model performance, we implemented 5-fold cross-validation where each fold was comprised only of samples from one of the 5 centers. This simulated a "real-life scenario" in which data from one or several centers are used for training and evaluating a model, that is then used in yet another center. For each cross-validation split of the data, we randomly set aside 20% of training samples to tune hyperparameters of the model and to assess the model performance during training.

Table 4.2: Segmentation results obtained on the different test folds with the use of cross-validation. The proposed model was compared to the standard U-Net model and the fixed thresholding method in terms of DSC, precision and recall. The mean and standard deviation of each metric on the test folds are computed across corresponding data samples. Average results are reported across the test folds.

Metrics	Model	Test fold					Average
		Brest (n=69)	Nantes (n=23)	Montreal (n=26)	Barcelona (n=24)	Liège (n=90)	
DSC	$T40$	0.33 ± 0.36	0.57 ± 0.41	0.37 ± 0.31	0.22 ± 0.24	0.18 ± 0.22	0.33 ± 0.15
	U-Net	0.68 ± 0.20	0.79 ± 0.12	0.77 ± 0.13	0.83 ± 0.10	0.79 ± 0.13	0.77 ± 0.05
	Ours	0.77 ± 0.15	0.81 ± 0.13	0.77 ± 0.21	0.84 ± 0.11	0.79 ± 0.13	0.80 ± 0.03
Precision	$T40$	0.30 ± 0.37	0.56 ± 0.43	0.29 ± 0.28	0.18 ± 0.21	0.14 ± 0.21	0.30 ± 0.16
	U-Net	0.61 ± 0.24	0.75 ± 0.15	0.74 ± 0.16	0.81 ± 0.17	0.79 ± 0.18	0.74 ± 0.08
	Ours	0.69 ± 0.20	0.73 ± 0.16	0.77 ± 0.22	0.81 ± 0.18	0.77 ± 0.19	0.75 ± 0.05
Recall	$T40$	0.48 ± 0.43	0.74 ± 0.35	0.65 ± 0.40	0.38 ± 0.36	0.38 ± 0.37	0.52 ± 0.17
	U-Net	0.88 ± 0.15	0.87 ± 0.10	0.85 ± 0.15	0.90 ± 0.09	0.84 ± 0.14	0.87 ± 0.02
	Ours	0.93 ± 0.10	0.96 ± 0.04	0.83 ± 0.19	0.91 ± 0.08	0.87 ± 0.14	0.90 ± 0.05

4.3.5 Evaluation Metrics

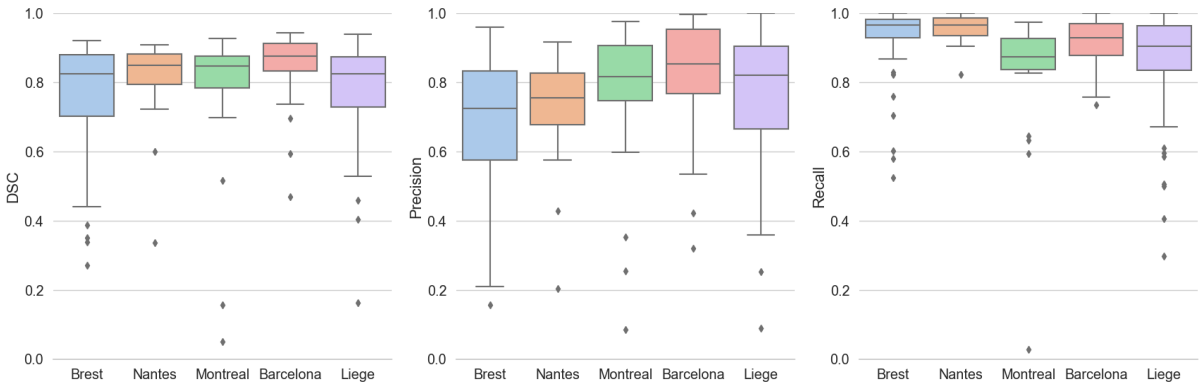
Aside from the DSC metric quantifying global volume overlap, we used precision (a.k.a. positive predictive value) P and recall R (a.k.a. sensitivity) to further evaluate model performance, as recommended by the TG211 [82], where DSC can be written as the harmonic mean of precision and recall:

$$DSC = 2\frac{P \cdot R}{P + R} \tag{4.2}$$

Using these metrics we compared our proposed network to the standard U-Net as a baseline model. In addition, a comparison was made with the use of a fixed thresholding method (based on 40% of the maximum standardized uptake within the tumor, denoted from here onward as T40), still widely used in the literature despite its obvious limitations [82, 83].

4.4 Results and Discussion

The results of all methods are summarised in Table 4.2 and Table A.1. As expected, T40 obtained poor performance across all test folds compared to U-Net and the proposed model. On average, our proposed model outperformed its U-Net counterpart in terms of DSC (0.80 vs 0.77), with a slightly smaller spread (0.03 vs 0.05). The largest difference between the proposed method and its standard counterpart was



(a) DSC (b) Precision (c) Recall

Figure 4.2: Box plots of the results on the test folds.

measured on the 'Brest' test fold, where U-Net demonstrated relatively poorer performance (0.77 vs 0.68). However, on the other test folds, both models achieved closer results. The superiority of the proposed model was due to a better recall (0.90 vs 0.87), whereas the difference in terms of precision was smaller (0.75 vs 0.74). Kolmogorov-Smirnov and Wilcoxon signed-rank tests both indicated that the difference in predictions of two models was significant ($\alpha = 0.05$) only for all evaluation metrics on the 'Brest' test fold, and for recall on 'Nantes' (see Table A.1).

This finding is in line with previous observations that, when properly tuned, the standard U-Net model can provide highly competitive results in many image segmentation tasks, especially in medical imaging. For example, top-ranked results were obtained in recent segmentation challenges using the ordinary U-Net model [109, 110, 113]. Under these circumstances, each step in the entire pipeline (e.g, data preprocessing, data augmentation, training procedure, etc.) may have a much larger impact on the model performance than a careful or complex re-design of the model architecture. For instance, we observed in our experiments that applying b-spline interpolation for image resampling instead of linear interpolation deteriorated both models performance by an average of nearly 8.5% on the test folds.

Both models achieved higher recall (between 0.83 and 0.93 on average) than precision (between 0.61 and 0.81) in all test folds, showing a trend in over-estimating the ground-truth rather than under-estimating it. One of the most challenging aspects pertaining to cervical cancer segmentation in PET images is to distinguish the tumor uptake from the adjacent bladder uptake. In all cases, even when the tumor was very close to the tumor, the proposed approach was capable to address this problem independently on the size, location and shape of the tumor uptake (see examples in

Figure 4.3). However, the wider spread of results on the two largest test folds (Brest, Liège) (Figure 4.2) could mean that the applied data augmentation techniques are not able to completely mimic all possible variations in presented PET images and alternatives should be investigated, such as, for example, relying on realistic simulated PET images to add more data for training. In addition, due to our 5-fold evaluation scheme based on clinical centers, the size of training sets (and as a result the variety of encountered examples) varied substantially (e.g., holding out Liège yields 146 training cases whereas holding out Nantes yields 209), which could also contribute in explaining these differences.

Analyzing predictions of the models, we identified a number of outliers[2] in each test fold (see examples in Figure 4.4). When considering the DSC metric, the total number of outliers in the entire dataset was equal to 15 (12 for U-Net) and varies from 2 to 4 across the test folds. In most cases, the model failed to accurately segment images with relatively small tumor regions (Figures 4.4a, 4.4d). More specifically, 11 outliers corresponded to cases where the tumor size was less than 200 voxels (i.e., 6.4 cm^3), whereas the average value across the dataset was 1160 voxels (i.e., 37.12 cm^3). The other source of errors in the model predictions is the presence of surrounding tissues with relatively high uptake that can be misclassified as the tumor (Figures 4.4b, 4.4c, 4.4e).

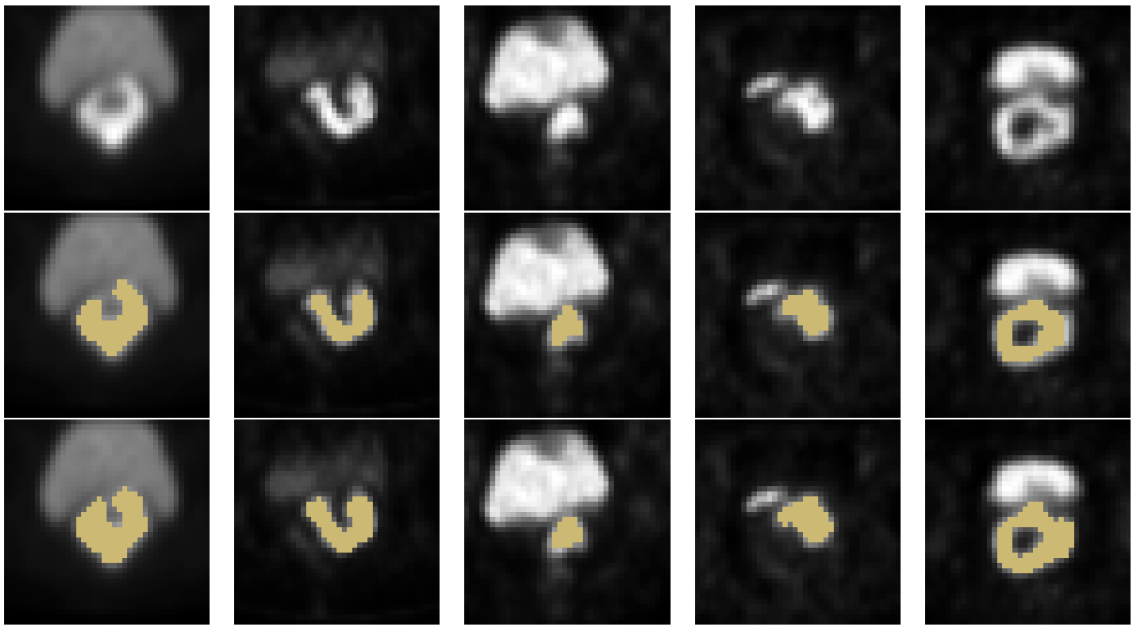
The performance was affected by tumor volume (see Table A.2 and Figure A.2): the lowest results were obtained in the first decile group[3] with DSC = 0.56 compared to the performance obtained on larger tumor volumes (significantly higher between 0.71 and 0.85). This happened due to precision that was steeply increasing along with the tumor size (0.44 to 0.90) whereas recall remained relatively stable (between 0.81 and 0.94). Examining the impact of the tumor contrast[4], we found that the proposed model demonstrated the worst results on low contrast images (see Table A.3 and Figure A.3). The decile group with the lowest tumor contrast had DSC = 0.67 and recall = 0.77, which were significantly different from the results on other groups (0.77 to 0.84, and 0.88 to 0.93, respectively). Investigating the relation between FIGO stages and the model performance, we did not find significant differences with DSC ranging from 0.75 to 0.82 (see Table A.4 and Figure A.4).

It should be emphasized on that although we used a previously well validated ap-

[2]A data point x_i from a dataset $X = \{x_1, \ldots, x_n\}$ is an *outlier*, if $x_i < q_1 + 1.5(q_3 - q_1)$, where q_i is the *i-th* empirical quartile of X.

[3]The first decile group corresponds to 10% of patients with the smallest tumors.

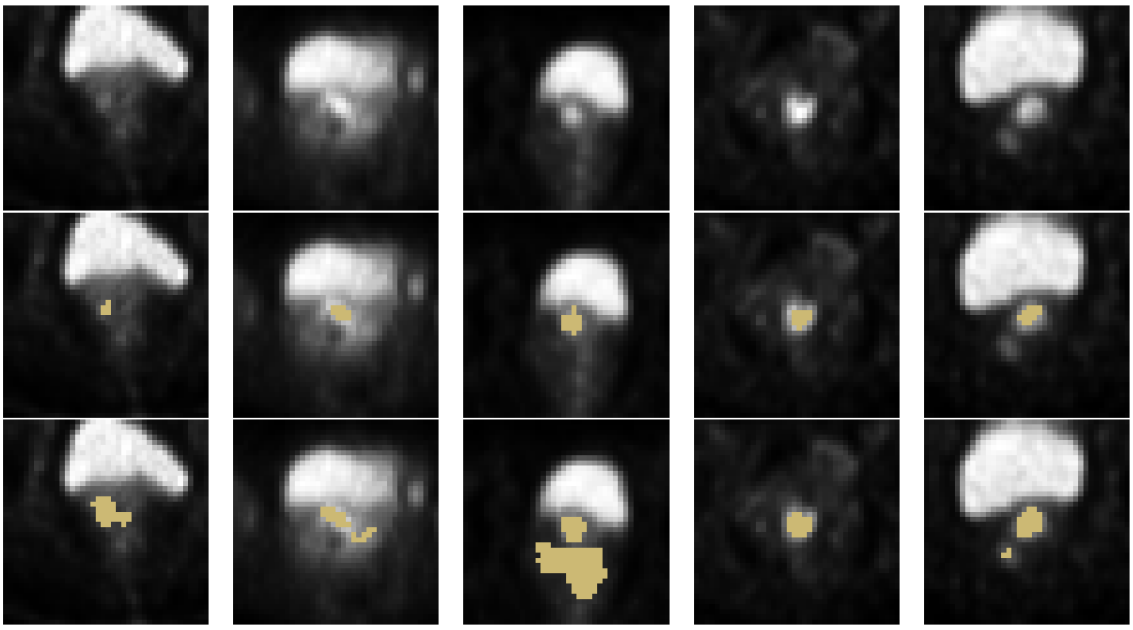
[4]We roughly estimated tumor contrast as the ratio between average tumor intensity and average intensity of the body region.



(a) Brest (0.80, 0.70, 0.95) (b) Nantes (0.84, 0.73, 0.98) (c) Montreal (0.81, 0.73, 0.90) (d) Barcelona (0.88, 0.82, 0.94) (e) Liège (0.81, 0.77, 0.87)

Figure 4.3: Examples of the model predictions on the test folds. Axial slices. (First row) Input images, (second row) input images with ground truth segmentation, (last row) input images with predicted segmentation. Evaluation metrics for whole scans are provided in format (DSC, precision, recall).

proach to define the ground-truth, this remains a surrogate of truth. In the absence of perfectly registered histopathological spatial information, this is the best we can achieve with a single segmentation method, which obviously provides imperfect results in a small number of cases (for instance highly heterogeneous or very small and low contrast cases) [83]. An even better approach would consist in generating several manual delineations by experts (at least three) in addition to the results of FLAB (other algorithms with proven good performance [83] could be added too) and generating a statistical consensus of all these segmentation results. This would provide the proposed model an even more reliable ground-truth to learn from, but it would be considerably more time-consuming and tedious, especially for generating the numerous manual delineations. Alternatively, our approach consisting in training the network on rigorously determined ground-truth masks could be reproduced by relying on other semi-automated methods with similar demonstrated levels of performance [82, 83]. Once trained, the proposed network can be applied to new data instantaneously, without the need for user intervention beyond checking and validating the



(a) Brest (0.34, 0.21, 0.87) (b) Nantes (0.34, 0.20, 0.97) (c) Montreal (0.16, 0.09, 0.90) (d) Barcelona (0.59, 0.42, 1.0) (e) Liège (0.40, 0.25, 1.0)

Figure 4.4: Examples of outliers in each test fold. Axial slices. (First row) Input images, (second row) input images with ground truth segmentation, (last row) input images with predicted segmentation. Evaluation metrics for whole scans are provided in format (DSC, precision, recall).

output result.

Unlike Chen et al. [32], we did not rely on any post-processing techniques built on prior anatomical information. First, based on the segmentation results of the proposed model, it appears able to natively learn the anatomic position of the tumor relative to the bladder from training samples without an additional prior guidance. Second, the assumption about the tumor roundness contradicts numerous examples in our dataset, especially these with heterogeneous distributions (Figures 4.3b, 4.3e). Although in the present case we focused on PET-only delineation, the proposed model can be trained using multiple different modalities as input. It might be beneficial in specific cases, such as dealing with small and/or low contrast tumors, to extract additional information from associated CT or MRI modalities. However, the main challenge in that case is to have a reliable ground-truth determined on fused multimodal data, which could prove quite difficult in the cervical region due to anatomical deformations and differences between PET and CT datasets.

With respect to our original objectives, our results obtained with the use of mul-

ticenter cross-validation allow to conclude that the designed model is able to provide similar performance on PET images from different institutions and is robust to variations in scanner types, reconstruction algorithms and post-processing methods. In addition, it allows for fully automated delineation of the tumor uptake without the need to exclude the bladder uptake, either manually or through the incorporation of additional prior information or constraints.

4.5 Conclusion

In this work we trained a modified U-Net model for fully automatic tumor uptake delineation in PET images in a multicenter context, without the need for additional anatomical information or prior constraints. The ability of the proposed model to learn and perform well for this task was demonstrated in PET images of 232 patients collected from five institutions. The ground-truth labels for all patient were generated by experts with the use of a semi-automated algorithm, to reduce observer-related variability and to avoid relying on manual delineations. We presented a versatile pipeline that includes appropriate data preprocessing and augmentation, design of the model architecture beyond the standard U-Net model, and an optimized training procedure. We mimicked a typical clinical scenario and conducted all experiments in a multicenter context. The designed model obtained good average accuracy for all considered institutions with very small standard deviation (DSC of 0.80 ± 0.03) without requiring any change in the pipeline. It slightly improved accuracy over the standard U-Net model although both approaches provided good results and largely outperformed the fixed threshold approach. The described approach managed to avoid including the bladder uptake in the resulting segmentation without the need for additional anatomical information (for instance, using the CT image) or priors such as shape constraints, and can therefore achieve fully automated delineation of the tumor uptake without the need for any user intervention. It can be implemented with minimal modifications to solve a variety of other segmentation tasks in different medical imaging modalities and could facilitate the deployment of fully automated radiomics pipelines.

Chapter 5

Delineation of Head and Neck Tumors in PET/CT

Reference

Squeeze-and-Excitation Normalization for Automated Delineation of Head and Neck Primary Tumors in Combined PET and CT Images. Andrei Iantsen, Dimitris Visvikis, and Mathieu Hatt. In:*Andrearczyk V., Oreiller V., Depeursinge A. (eds) Head and Neck Tumor Segmentation. HECKTOR 2020. Lecture Notes in Computer Science, vol 12603. Springer, Cham.* (2020).

Abstract

Development of robust and accurate fully automated methods for medical image segmentation is crucial in clinical practice and radiomics studies. In this work, we contributed an automated approach to head and neck (H&N) primary tumor segmentation in combined positron emission tomography / computed tomography (PET/CT) images in the context of the MICCAI 2020 Head and Neck Tumor segmentation challenge (HECKTOR). Our model was designed on the U-Net architecture with residual layers and supplemented with Squeeze-and-Excitation Normalization. The described method achieved competitive results in cross-validation (DSC 0.745, precision 0.760, recall 0.789) performed on different centers, as well as on the test set (DSC 0.759, precision 0.833, recall 0.740) that allowed us to

win first prize in the HECKTOR challenge among 64 registered teams. The full implementation based on PyTorch and the trained models are available at https://github.com/iantsen/hecktor.

This chapter describes a CNN-based approach that was proposed by our team to address the task of H&N tumor segmentation in the context of the HECKTOR challenge. A part of this chapter repeats exactly our previously published findings (see the paper cited above). However, a large amount of supplementary information was included relying on a number of overview papers published after the end of the challenge. Apart from a few minor modifications (in HECKTOR, dropout layers were not used for regularization), the model described in this chapter is identical to the one presented in Chapter 6 for a task of brain tumor segmentation in MRI scans. Moreover, the described pipelines (i.e., data preprocessing, augmentation, training procedures, etc.) vary insignificantly and mainly because of the difference in input image modalities. Nevertheless, the described approach obtained highly competitive results in both competitions, essentially without any task-specific adjustments.

5.1 Introduction

As stated earlier in Chapter 4, combined PET/CT imaging is widely used in clinical practice, for instance, for radiotherapy treatment planning, initial staging and response assessment. PET and CT modalities provide complementary information on metabolic and morphological tissue properties and therefore can be used for malignant lesion segmentation. In radiomics, quantitative evaluation of radiotracer uptake in PET and tissue density in CT aims at extracting clinically relevant features in order to build diagnostic and prognostic tools. However, the segmentation stage in the radiomics pipeline represents the significant bottleneck, since it is a tedious and time-consuming process that typically suffers from a high observer-related variability, especially if manual segmentation is used as a ground-truth. Under these circumstances, more efficient methods are highly desirable to automate the segmentation process and facilitate its clinical routine usage.

The prime focus of the MICCAI 2020 Head and Neck Tumor segmentation challenge (HECKTOR) [5, 167] is on evaluating automatic algorithms for head and neck (H&N) tumor segmentation in combined PET and CT images. In the context of this challenge, all participants are asked to design an approach to segment Gross Tumor

Table 5.1: Summary of the HECKTOR dataset.

Center	Numb. of patients	Split	Scanner
Hôpital général Juif, Canada (HGJ)	55	Train	GE Discovery ST
Centre Hospitalier Universitaire de Sherbooke, Canada (CHUS)	72	Train	Philips GeminiGXL
Hôpital Maisonneuve-Rosemont, Canada (HMR)	18	Train	GE Discovery STE
Centre Hospitalier de l'Université de Montréal, Canada (CHUM)	56	Train	GE Discovery STE
Centre Hospitalier Universitaire Vaudois, Switzerland (CHUV)	53	Test	GE Discovery D690

Volume of the primary tumor (GTVt) in provided images of patients with oropharyngeal cancer. Since some part of the dataset was acquired for the whole body, each data example is accompanied by a bonding box to detect the oropharynx region. In total, a training dataset consisting of 201 patients from four medical centers (HGJ, CHUS, HMR and CHUM) located in Québec, Canada, is available for model development. A test set comprised of 53 patients without ground-truth labels from a different center in Switzerland (CHUV) is used for assessment (see Table 5.1). As a result, all images were acquired using many different systems and acquisition protocols (see details in [5, 167]). The training dataset consists of images initially presented in [217] that were manually re-annotated by an expert for the purpose of the challenge. All challenge participants have up to five attempts to submit their predictions for the test set. The Dice similarity coefficient (DSC), precision and recall metrics are computed for each submission. However, the final ranking is based only on the average DSC across examples in the test set.

This chapter describes our approach based on convolutional neural networks (CNNs) supplemented with Squeeze-and-Excitation Normalization (SE Normalization or SE Norm) layers to address the goal of the HECKTOR challenge.

5.2 Materials and Methods

5.2.1 SE Normalization

The key element of our model is SE Normalization layers [105, 106] that we recently proposed in the context of the MICCAI 2020 Brain Tumor Segmentation

(BraTS) challenge [13, 14, 156]. Similarly to instance normalization [215], for an input $\mathrm{X} = (x_1, x_2, \ldots, x_\mathrm{C})$ with C channels, SE Norm layer first normalizes all channels of each example in a batch using the mean and standard deviation:

$$x'_i = \frac{1}{\sigma_i}(x_i - \mu_i)\,, \tag{5.1}$$

where $\mu_i = \mathrm{E}[x_i]$ and $\sigma_i = \sqrt{\mathrm{Var}[x_i] + \epsilon}$ with ϵ as a small constant to prevent division by zero. After, a pair of parameters γ_i, β_i are applied to each channel to scale and shift the normalized values:

$$y_i = \gamma_i x'_i + \beta_i\,. \tag{5.2}$$

In case of instance normalization, both parameters γ_i, β_i, are fitted in the course of training, stay fixed and independent on the input X during inference. By contrast, we propose to model the parameters γ_i, β_i as functions of the input X by means of Squeeze-and-Excitation (SE) blocks [99], i.e.,

$$\begin{aligned} \gamma &= f_\gamma(X)\,, \\ \beta &= f_\beta(X)\,, \end{aligned} \tag{5.3}$$

where $\gamma = (\gamma_1, \gamma_2, \ldots, \gamma_\mathrm{C})$ and $\beta = (\beta_1, \beta_2, \ldots, \beta_\mathrm{C})$ - the scale and shift parameters for all channels, f_γ - the original SE block with the sigmoid activation, and f_β is modeled as the SE block with the tanh activation function to enable the negative shift (see Figure 5.1a). Both SE blocks first apply global average pooling (GAP) to squeeze each channel into a single vector, a descriptor. Then, the descriptor is passed through two fully connected (FC) layers to capture non-linear cross-channel dependencies. The first FC layer is implemented with the reduction ratio r to form a bottleneck for controlling model complexity. Throughout this chapter, we apply SE Norm layers with the fixed reduction ration $r = 2$.

5.2.2 Network Architecture

Our model is built upon a seminal U-Net architecture [34, 183] with the use of SE Norm layers [105, 106]. Convolutional blocks that form the model decoder are stacks of $3 \times 3 \times 3$ convolutions and ReLU nonlinearity followed by SE Norm layers. Residual blocks in the encoder consist of convolutional blocks with shortcut connections (see Figure 5.1b). If the number of input/output channels in a residual block is different,

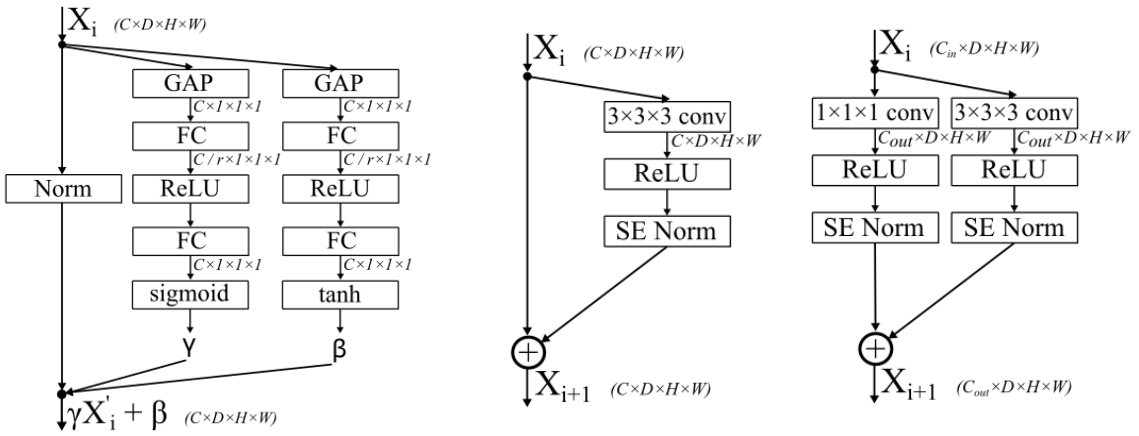


Figure 5.1: Layers with SE Normalization: (a) SE Norm layer, (b) residual layer with the shortcut connection, and (c) residual layer with the non-linear projection. Output dimensions are depicted in italics.

a non-linear projection is performed by adding a $1 \times 1 \times 1$ convolutional block to the shortcut in order to match the dimensions (see Figure 5.1c).

In the encoder, downsampling is performed by applying max pooling with the kernel size of $2 \times 2 \times 2$. To linearly upsample feature maps in the decoder, $3 \times 3 \times 3$ transposed convolutions are used. In addition, we supplement the decoder with three upsampling paths to transfer low-resolution features further in the model by applying a $1 \times 1 \times 1$ convolutional block to reduce the number of channels, and utilizing trilinear interpolation to increase the spatial size of the corresponding feature maps (see Figure 5.2, yellow blocks).

The first residual block placed right after the input is implemented with a $7 \times 7 \times 7$ kernel to increase the receptive field of the model without significant computational overhead. The sigmoid function is applied after the last block to generate probabilities for two target classes.

5.2.3 Data Preprocessing and Sampling

Both PET and CT images were first resampled to a common resolution of $1 \times 1 \times 1$ mm^3 with trilinear interpolation. Each training example was a patch of $144 \times 144 \times 144$ voxels randomly extracted from a whole PET/CT image, whereas validation examples were received from the bounding boxes provided by organizers. Training patches were extracted to include the tumor class with a 0.9 probability to reduce class imbalance and facilitate model training.

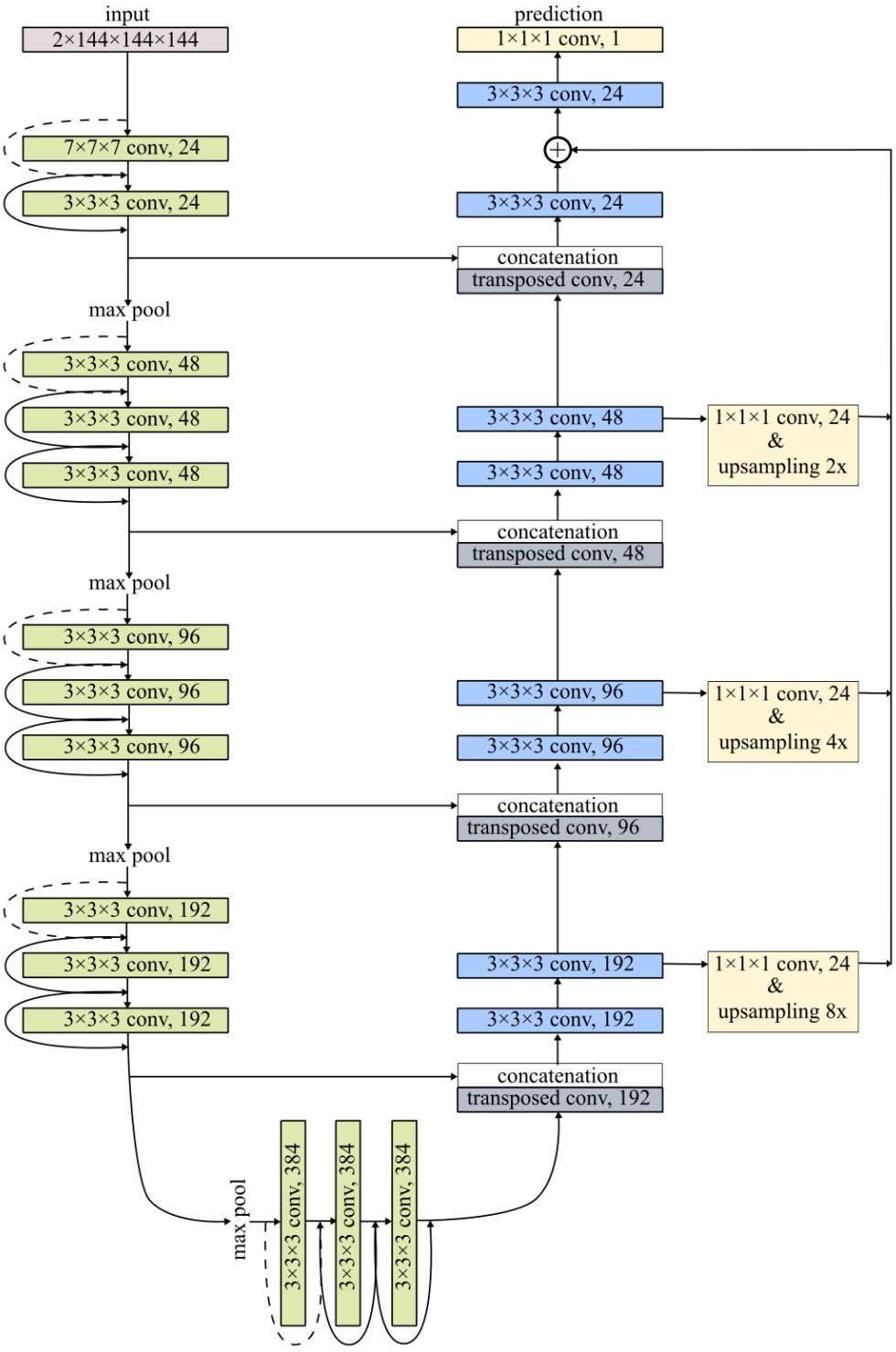


Figure 5.2: The model architecture with SE Norm layers. The input consists of PET/CT patches of the size of $144 \times 144 \times 144$ voxels. The encoder consists of residual blocks with identity (solid arrows) and projection (dashed arrows) shortcuts. The decoder is formed by convolutional blocks. Additional upsampling paths are added to transfer low-resolution features further in the decoder. Kernel sizes and numbers of output channels are depicted in each block.

CT intensities were first clipped in the range of $[-1024, 1024]$ Hounsfield Units and then mapped to $[-1, 1]$. PET images were transformed independently with the use of Z-score normalization, performed on each patch.

5.2.4 Training Procedure

The model was trained for 800 epochs using Adam optimizer on two GPUs NVIDIA GeForce GTX 1080 Ti (11 GB) with a batch size of 2 (one sample per worker). The cosine annealing schedule was applied to reduce the learning rate from 10^{-3} to 10^{-6} within every 25 epochs.

5.2.5 Loss Function

The unweighted sum of the Soft Dice Loss [158] and the Focal Loss [139] was employed for training. Based on [158], the Soft Dice Loss for one data example can be written as

$$L_{Dice}(y, \hat{y}) = 1 - \frac{2\sum_i^{\mathrm{N}} y_i \hat{y}_i + 1}{\sum_i^{\mathrm{N}} y_i^2 + \sum_i^{\mathrm{N}} \hat{y}_i^2 + 1}\,. \tag{5.4}$$

The Focal Loss is defined as

$$L_{Focal}(y, \hat{y}) = -\frac{1}{\mathrm{N}} \sum_i^{\mathrm{N}} y_i (1 - \hat{y}_i)^{\gamma} \log(\hat{y}_i)\,. \tag{5.5}$$

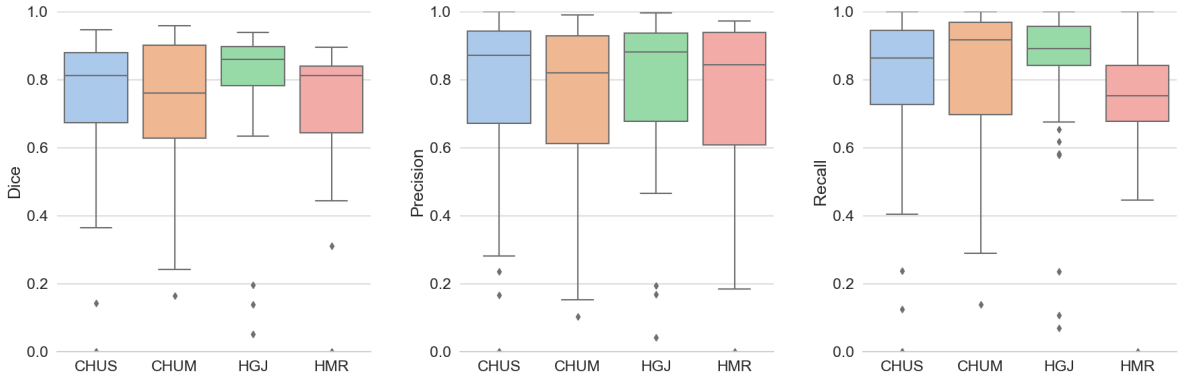
In both definitions, $y_i \in \{0, 1\}$ - the label for the *i-th* voxel, $\hat{y}_i \in [0, 1]$ - the predicted probability for the *i-th* voxel, and N - the total numbers of voxels. Also, we add $+1$ to the numerator and denominator in the Soft Dice Loss to avoid a division by zero in cases where the tumor class is not present in training patches. The parameter γ in the Focal Loss is set at 2.

5.2.6 Ensembling

Results on the test set were produced with the use of an ensemble of eight models trained and validated on different splits of the training set. Four models were built using multicenter cross-validation, i.e, the data from three centers was used for training while the data from the fourth center was held out for validation. Four additional models were fitted on random data splits in order to take into account potential, center-specific differences in data distributions caused by variations in scanners and

Table 5.2: Results on different cross-validation splits. Average results (the row 'Average') are provided for each evaluation metric across all centers in the context of multicenter cross-validation (first four rows). The mean and standard deviation of each metric are computed across all data samples in the corresponding validation center. The row 'Average (rs)' indicates the average results on the four random data splits.

Center	DSC	Precision	Recall
CHUS ($n = 72$)	0.744 ± 0.206	0.763 ± 0.248	0.788 ± 0.226
CHUM ($n = 56$)	0.739 ± 0.190	0.748 ± 0.224	0.819 ± 0.216
HGJ ($n = 55$)	0.801 ± 0.180	0.791 ± 0.208	0.839 ± 0.200
HMR ($n = 18$)	0.696 ± 0.232	0.739 ± 0.286	0.712 ± 0.228
Average	0.745	0.760	0.789
Average (rs)	0.757	0.762	0.820



(a) DSC (b) Precision (c) Recall

Figure 5.3: Distributions of the results on multicenter cross-validation splits.

acquisition protocols. Examples in these splits were sampled randomly with stratification to preserve the original percentage of examples for each center. Predictions on the test set were produced by averaging predictions of the individual models and applying a threshold operation with a value equal to 0.5.

5.3 Results and Discussion

Our cross-validation results in the context of the HECKTOR challenge are summarized in Table 5.2. The best outcome in terms of all evaluation metrics was received for the 'HGJ' center with 55 patients (DSC of 0.801, precision of 0.791, and recall of 0.712). The model demonstrated the lowest performance for the 'HMR' center that is

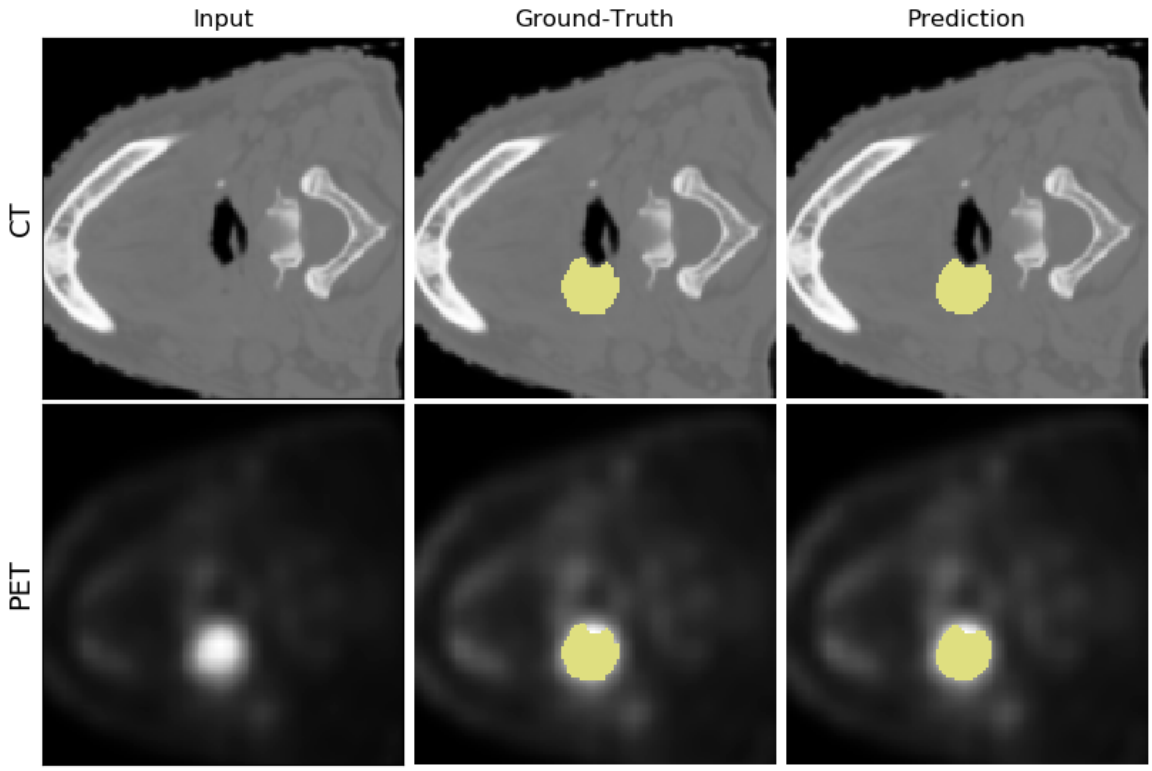


Figure 5.4: Example of high-quality predictions on multicenter cross-validation splits. Patient 'CHUM062' with DSC = 0.924, precision = 0.940 and recall = 0.909.

the least represented in the whole dataset. The difference with the two other centers was minor for all evaluation metrics. Relatively small spreads between each center and the average results imply that the model predictions were robust enough, even without any center-specific standardization. This finding is supported by the lack of significant differences in the average DSC between the multicenter and random cross-validation splits (0.745 vs 0.757).

Apart from the 'HMR' center, the model demonstrated higher recall than precision for all centers, tending to slightly over-estimate the target volume (recall 0.789 vs precision 0.760). In case of random validation splits, this difference even increased (recall 0.82 vs precision 0.762), although precision remained practically unchanged. This might be due to a small number of hard, non-representative cases that significantly affect the model performance, depending on whether they are used for training or validation. For instance, the poor model performance can be caused by the low amount of radiotracer uptake that makes lesions barely visible in some PET images (see Figure 5.5). Therefore, it is necessary to have a sufficient number of such examples in the training sample to alleviate this problem. In addition, according to

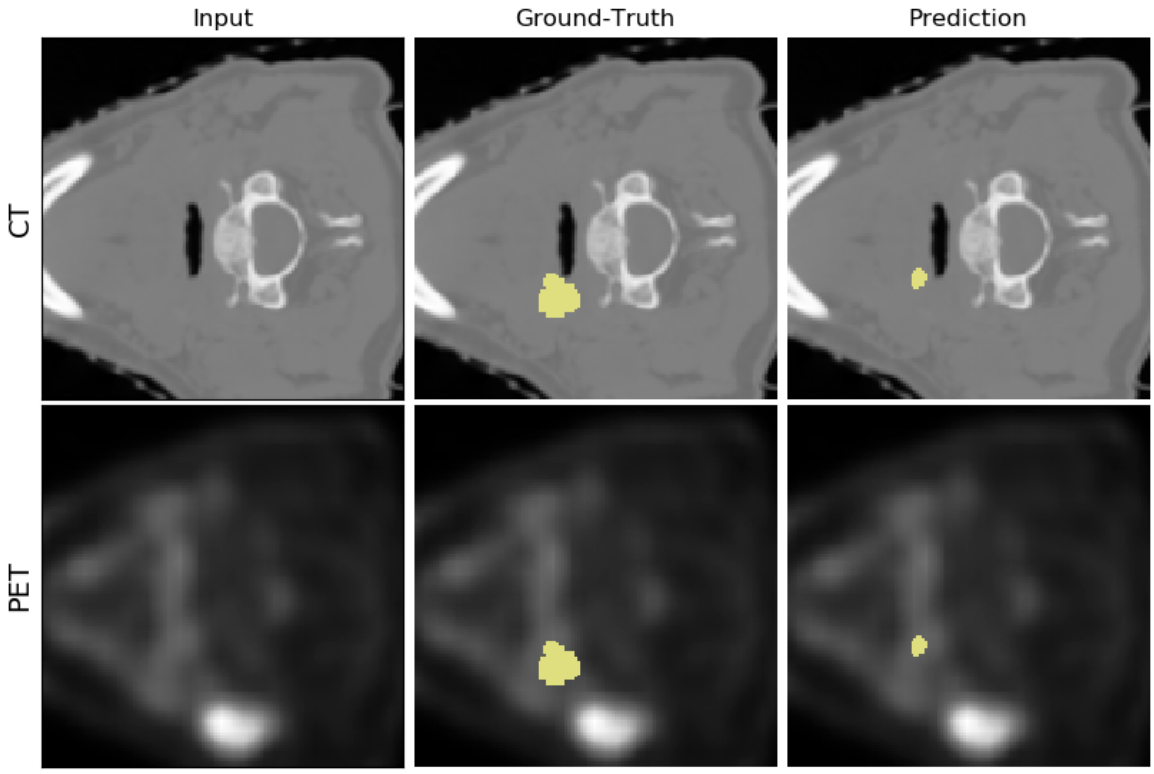


Figure 5.5: Example of low-quality predictions on multicenter cross-validation splits. Patient 'CHUS026' with DSC = 0.364, precision = 0.780 and recall = 0.238

.

Andrearczyk et al. [5] and Oreiller et al. [167], CNNs tend to demonstrate poor results in the context of the HECKTOR challenge in cases where primary tumors look like lymph nodes, there exists abnormal uptake in the tongue, or lesions are located at the border of the oropharynx region. These challenging examples are presented in every center, and most of them belong to the 'HGJ' center (see Figure 5.3).

Despite the hard cases, the model achieved highly accurate results for the vast majority of patients (the median DSC of 0.813) in multicenter cross-validation (see Figure 5.6). Moreover, it was able to correctly segment diverse cases combining complementary information from both image modalities. For example, the predicted contour containing the area with high FDG uptake was corrected by the model using the CT scan to exclude air in the trachea (see Figure 5.4).

The ensemble results on the test set consisting of 53 patients from the 'CHUV' center are summarized in Table 5.3. On the previously unseen data, the ensemble of eight models achieved the highest result among all participating teams with DSC of 75.9%, precision of 83.3% and recall of 74%. Our approach significantly outperformed both baseline methods provided by the challenge organizers ('Baseline 2D'

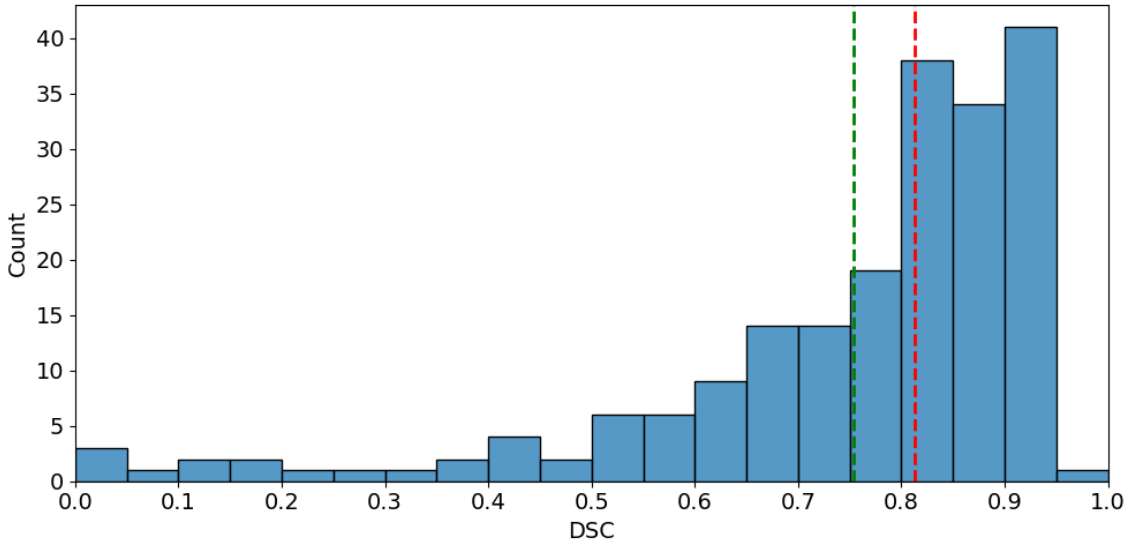


Figure 5.6: Histogram of the DSC distribution across all patients in multicenter cross-validation with the mean = 0.754 (green) and median = 0.813 (red)

Table 5.3: Summary of the challenge results. The average DSC, precision, and recall are reported for the top-5 teams and two baseline models. The final ranking is based on the average DSC across examples in the test. See details in Andrearczyk et al. [6]

Team	DSC	Precision	Recall	Rank
andrei.iantsen (ours)	0.759	0.833	0.740	1
junma	0.752	0.838	0.717	2
badger	0.735	0.833	0.702	3
deepX	0.732	0.785	0.732	4
AIView_sjtu	0.724	0.848	0.670	5
Baseline 3D	0.661	0.591	0.853	–
Baseline 2D	0.659	0.624	0.763	–

and 'Baseline 3D' in Table 5.3). Although our average results on the test set are higher, in terms of recall, than the results of the second best participant (0.740 vs 0.717), the difference in DSC between our teams is statistically insignificant. However, the difference with all other participants is significant [5, 167]. In addition, the organizers provided an estimate for inter-observer agreement between four different experts, which was measured on a random data subset of 21 patients. The average DSC metric calculated for all possible pairs of these experts was 0.61, which is considerably worse than the results of most participants. The relatively low value of inter-observer agreement is partly due to the lack of clear clinical guidelines for GTVt segmentation in combined PET/CT images. However, it also demonstrates

the high potential of CNN-based models to surpass human experts in this task and be subsequently integrated in clinical practice.

5.4 Conclusion

In this work we presented the CNN-based approach with a new type of layers, referred to as SE Normalization, to address the task of the H&N tumor segmentation in the context of the HECKTOR challenge. The ability of our method to provide accurate segmentation for tumor lesions in PET/CT images was demonstrated with the use of multicenter cross-validation and the independent test set. Our approach obtained the best results in terms of DSC among all participating teams, and hopefully took one more step towards the integration of CNN-based methods into daily clinical practice.

Chapter 6

Brain Tumor Segmentation in Multisequence MRI

Reference

Squeeze-and-Excitation Normalization for Brain Tumor Segmentation.. Andrei Iantsen, Vincent Jaouen, Dimitris Visvikis, and Mathieu Hatt. In:*Crimi A., Bakas S. (eds) Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries. BrainLes 2020. Lecture Notes in Computer Science, vol 12659. Springer, Cham.* (2021).

Abstract

In this paper we described our approach for glioma segmentation in multisequence magnetic resonance imaging (MRI) in the context of the MICCAI 2020 Brain Tumor Segmentation Challenge (BraTS). We proposed an architecture based on U-Net with a new computational unit termed "SE Norm" that brought significant improvements in segmentation quality. Our approach obtained competitive results on the validation (Dice scores of 0.780, 0.911, 0.863) and test (Dice scores of 0.805, 0.887, 0.843) sets for the enhanced tumor, whole tumor and tumor core sub-regions.

6.1 Introduction

Glioma is a group of malignancies that arises from the glial cells in the brain. Nowadays, gliomas are the most common primary tumors of the central nervous system [13, 216]. The symptoms of patients presenting with a glioma depend on the anatomical site of the glioma in the brain and can be too common (e.g., headaches, nausea or vomiting, mood and personality alterations, etc.) to give an accurate diagnosis in early stages of the disease. The primary diagnosis is usually confirmed by magnetic resonance imaging (MRI) or computed tomography (CT) that provide additional structural information about the tumor.

Gliomas usually consist of heterogeneous sub-regions (i.e., peritumoral edematous/invaded tissue, necrotic core, active and non-enhancing core) with variable histologic and genomic phenotypes [12, 156]. This heterogeneity of gliomas is also reflected in their imaging phenotype, as their sub-regions are described by varying intensity profiles in MRI scans, indicating varying tumor biological properties. Due to its ability to depict the tumor sub-regions with different intensities, multimodal MR imaging is routinely used for non-invasive tumor evaluation and treatment planning. However, manual detection and delineation of tumor sub-regions is tedious, time-consuming and subjective because of the high heterogeneity in tumor appearances and shapes. In clinical settings, this manual process is routinely carried out by radiologists in a qualitative visual manner and hence becomes impractical when dealing with numerous patients.

The BraTS challenge [13, 14, 156], running since 2012, is aimed at the development of automatic methods for brain tumor segmentation in MRI scans. The challenge participants are called to address this task by using a provided clinically-acquired training data to develop their method and produce segmentation labels of the glioma sub-regions. Additional tasks in the BraTS 2020 challenge, namely overall survival prediction and uncertainty estimation for the predicted tumor sub-regions, are beyond the scope of this research.

6.2 BraTS Challenge

6.2.1 Dataset

All participants of the BraTS challenge are provided with the clinically-acquired training dataset of pre-operative MRI scans (four sequences per patient) and segmentation masks for three different tumor sub-regions. The exact MRI data consists of 1) a na-

tive T1-weighted scan (T1), 2) a post-contrast T1-weighted scan (T1Gd), 3) a native T2-weighted scan (T2), and 4) a T2 Fluid Attenuated Inversion Recovery (T2-FLAIR) scan. This collection of brain tumor MRI scans was acquired from multiple different centers under standard clinical conditions but with different equipment and imaging protocols, resulting in a vastly heterogeneous image quality, reflecting diverse clinical practice across different institutions.

The dataset was segmented manually, by one to four raters in each center, following the same annotation protocol. This protocol was designed by the challenge organizers in order to make it possible to create similar ground-truth delineations across various annotators. The exact annotated regions were based upon known observations visible to the trained radiologist and comprised of the Gd-enhancing tumor (ET), the peritumoral edematous/invaded tissue (ED) and the necrotic tumor core (NCR). ET is the enhancing portion of the tumor, described by areas with both visually avid as well as faint enhancement on T1Gd MRI. NCR is the necrotic core of the tumor, the appearance of which is hypointense on T1Gd MRI. ED is the peritumoral edematous and infiltrated tissue, defined by the abnormal hyperintense signal envelope on the T2 FLAIR volumes, which includes the infiltrative non enhancing tumor as well as vasogenic edema in the peritumoral region. See Figure 6.1 for details.

The ground-truth annotations were only approved by domain experts whereas they were actually created by multiple experts. Although a very specific annotation protocol was provided to each data contributing institution, slightly different annotation styles were noted for the various raters involved in the process. Therefore, all final labels included in the BraTS dataset were also further reviewed for consistency and compliance with the annotation protocol by a single board-certified neuro-radiologist with more than 15 years of experience.

The provided data was distributed after its harmonization, following standardization preprocessing without affecting the apparent information in the images. Specifically, the preprocessing routines applied in all the BraTS MRI scans included co-registration to the same anatomical template, interpolation to a uniform isotropic resolution of 1 mm^3 and skull-stripping (see details in Menze et al. [156]).

The BraTS challenge was evolving over the years with a continuously increasing number of patient cases as well as through an improvement of the data split, used for algorithmic development and evaluation. In the BraTS 2020, the dataset of 660 patients is divided in training ($n = 369$), validation ($n = 125$), and testing ($n = 166$) datasets. The challenge participants are provided with the ground-truth labels only for the training data. The validation data is then provided to the participants without

any associated ground-truth and the testing data is kept hidden from the participants at all times.

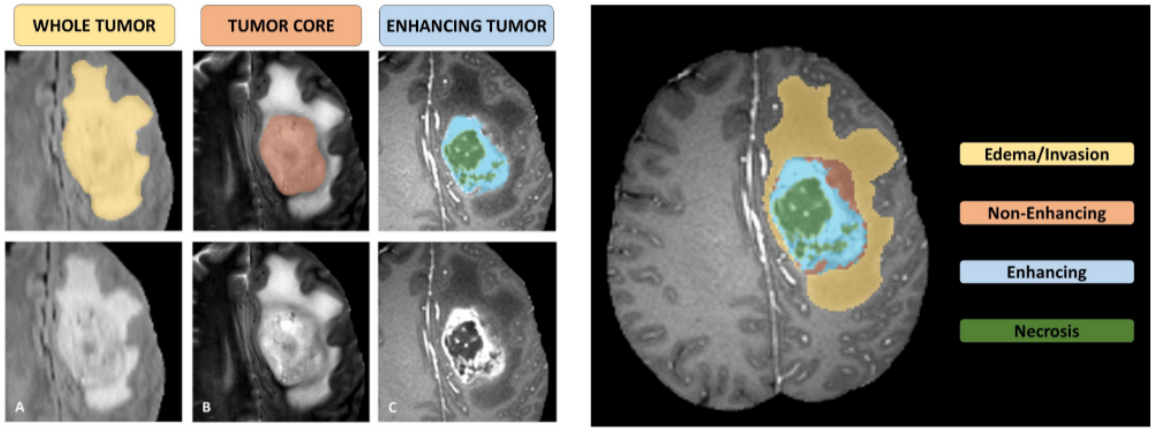


Figure 6.1: Glioma sub-regions. Image patches with the tumor sub-regions annotated in the different MRI modalities. The image patches show from left to right: the whole tumor (WT - yellow) visible in T2-FLAIR **(A)**, the tumor core (TC - orange) visible in T2 **(B)**, the enhancing tumor (ET - light blue) visible in T1-Gd, surrounding the cystic/necrotic components of the core (green) **(C)**. The segmentation masks are combined to generate the final labels of the tumor sub-regions **(D)**: edema/invasion (yellow), non-enhancing solid core (orange), necrotic/cystic core (green), enhancing core (blue). Source: Menze et al. [156]

6.2.2 Challenge Task

The challenge participants are called to develop automatic methods for brain tumor segmentation by using the provided training data. For each patient, the output must represent a corresponding segmentation mask for the target classes (i.e, ET, NCR and ED sub-regions). However, the following sub-regions (see Figure 6.1) are used for performance evaluation:

- "enhancing tumor" (ET), that corresponds to the ET sub-region;
- "tumor core" (TC), that describes the bulk of the tumor, which is what is typically considered for surgical excision. The TC entails the ET as well as the NCR parts of the tumor;
- "whole tumor" (WT), that entails the TC and ED parts, and describes the complete extent of the disease.

6.2.3 Performance Evaluation

Metrics

Results of the segmentation task are evaluated using the Dice similarity coefficient (DSC) and the Hausdorff distance (HD). Suppose T and P denote a ground-truth binary mask for a particular class and a binary prediction for this class, respectively. The DSC is computed as follows:

$$\mathrm{DSC}(\mathrm{T},\mathrm{P}) \stackrel{\mathrm{def}}{=\!=} \frac{2|\mathrm{T}\cap\mathrm{P}|}{|\mathrm{T}|+|\mathrm{P}|}\,, \tag{6.1}$$

where $|\cdot|$ is the total class size (i.e, the number of non-zero elements). The DSC metric quantifies the overlap between two segmentation mask.

The HD metric evaluates the maximum surface distance between two segmentation masks. It is defined as

$$\mathrm{d_{HD}}(\mathrm{T},\mathrm{P}) \stackrel{\mathrm{def}}{=\!=} \max\left\{\sup_{t\in\partial\mathrm{T}}\inf_{p\in\partial\mathrm{P}}\mathrm{d}(t,p),\ \sup_{p\in\partial\mathrm{P}}\inf_{t\in\partial\mathrm{T}}\mathrm{d}(p,t)\right\}, \tag{6.2}$$

where $\mathrm{d}(t,p)$ corresponds to the Euclidean distance between two points, t and p, lying on the ground-truth surface $\partial\mathrm{T}$ and the predicted surface $\partial\mathrm{P}$, respectively. This metric is generally sensitive to outliers, therefore the 95th percentile of the surface distance is often used instead. Throughout this chapter, we in fact refer to the 95th percentile of the surface distance as the "Hausdorff distance".

Ranking Scheme

The BraTS ranking scheme assigns ranks to each team relative to its competitors for each of the testing subjects, for each evaluated region (i.e., ET, TC, WT), and for each metric (i.e., DSC and HD). In BraTS 2020, each team is ranked for 166 subjects, for 3 sub-regions, and for 2 metrics, which results in $166 \times 3 \times 2 = 996$ individual rankings. The final ranking score (FRS) for each team is then calculated by firstly averaging across all these individual rankings for each patient (i.e., cumulative rank), and then averaging these cumulative ranks across all patients for each participating team.

Then, permutation testing is conducted to determine statistical significance of the relative rankings between each pair of teams. This permutation testing reflects differences in performance that exceed those that might be expected by chance. More specifically, the challenge organizers start with a list of observed subject-level cumu-

lative ranks, i.e., the actual ranking described above, for each team. For each pair of teams, they repeatedly randomly permute (for 100,000 times) the cumulative ranks for each subject. For each permutation, they calculate the difference in the FRS between this pair of teams. The proportion of times the difference in FRS, calculated using randomly permuted data, exceeds the observed difference in FRS (i.e., using the actual data), so it indicates the statistical significance of their relative rankings as a p-value. These values provide insights of statistically significant differences across each pair of participating teams (see details in Bakas et al. [12]).

6.3 Method

6.3.1 Network Architecture

The 3D U-Net [34, 183] serves as the basis to design our model. The basic element of the model, a convolutional block comprised of a $3 \times 3 \times 3$ convolution followed by the ReLU activation function and the SE Norm layer (described in Section 5.2.1), is used to construct the decoder (Figure 6.2, blue blocks). In the encoder, we utilize residual layers [89, 91] consisting of convolutional blocks with shortcut connections (Section 5.2.1, Figure 5.1b). If numbers of input/output channels in a residual layer are different, we perform a non-linear projection by adding a $1 \times 1 \times 1$ convolutional block to the shortcut in order to match the dimensions (Section 5.2.1, Figure 5.1c).

In the encoder, we perform downsampling applying max pooling with the kernel size of $2 \times 2 \times 2$. To linearly upsample feature maps in the decoder, we use $3 \times 3 \times 3$ transposed convolutions. In addition, we supplement the decoder with three upsampling paths to transfer low-resolution features further in the model by applying a $1 \times 1 \times 1$ convolutional block to reduce the number of channels, and utilizing trilinear interpolation to increase the spatial size of the feature maps (Figure 6.2, yellow blocks).

The first residual layer located after the input is implemented with the kernel size of $7 \times 7 \times 7$ to increase the receptive field of the model without significant computational overhead. The softmax layer is applied to output probabilities for four target classes.

To regularize the model, we add Spatial Dropout layers[1] [212] right after the last residual block at each resolution stage in the encoder and before the $1 \times 1 \times 1$

[1]The model introduced in Section 5 is built without the dropout layers, which is the main difference between presented architectures. Here, these layers are applied to achieve stronger regularization.

convolution in the decoder (Figure 6.2, red blocks).

6.3.2 Data Preprocessing

Intensities of MRI scans are not standardized and typically exhibit a high variability in both intra- and inter-image domains. In order to decrease the intensity inhomogeneity, we perform Z-score normalization for each MRI sequence and each patient separately. The mean and standard deviation are calculated based only on non-zero voxels corresponding to the brain region. All background voxels remain unchanged after normalization.

6.3.3 Training Procedure

Due to the large size of provided MRI scans, we perform training on random patches with a size of $144 \times 160 \times 192$ voxels ($depth \times height \times width$) on two GPUs NVIDIA GeForce GTX 1080 Ti (11 GB) with a batch size of 2 (one sample per worker).

We train the model for 300 epochs using Adam optimizer [122] with $\beta_1 = 0.9$ and $\beta_2 = 0.99$ for exponential decay rates for moment estimates, and apply a cosine annealing schedule [146] gradually reducing the learning rate from $lr_{max} = 10^{-4}$ to $lr_{min} = 10^{-6}$ within 25 epochs and performing the learning rate adjustment at each epoch.

6.3.4 Loss Function

We utilize the unweighted sum of the Soft Dice Loss [158] and the Focal Loss [139] as the composite loss function in the course of training. The Soft Dice Loss is the differentiable surrogate to optimize the DSC metric, that is one of the evaluation metrics used in the challenge. The Focal Loss, compared to the Soft Dice Loss, has much smoother optimization surface that ease the model training.

The Soft Dice Loss for a single training example can be written as

$$L_{Dice}(y, \hat{y}) = 1 - \frac{1}{\mathrm{C}} \sum_{c=1}^{\mathrm{C}} \frac{2 \sum_i^{\mathrm{N}} y_i^c \hat{y}_i^c + 1}{\sum_i^{\mathrm{N}} y_i^c + \sum_i^{\mathrm{N}} \hat{y}_i^c + 1} \,. \tag{6.3}$$

The Focal Loss is computed as

$$L_{Focal}(y, \hat{y}) = -\frac{1}{\mathrm{N}} \sum_{i}^{\mathrm{N}} \sum_{c=1}^{\mathrm{C}} y_i^c (1 - \hat{y}_i^c)^\gamma \log(\hat{y}_i^c) \,. \tag{6.4}$$

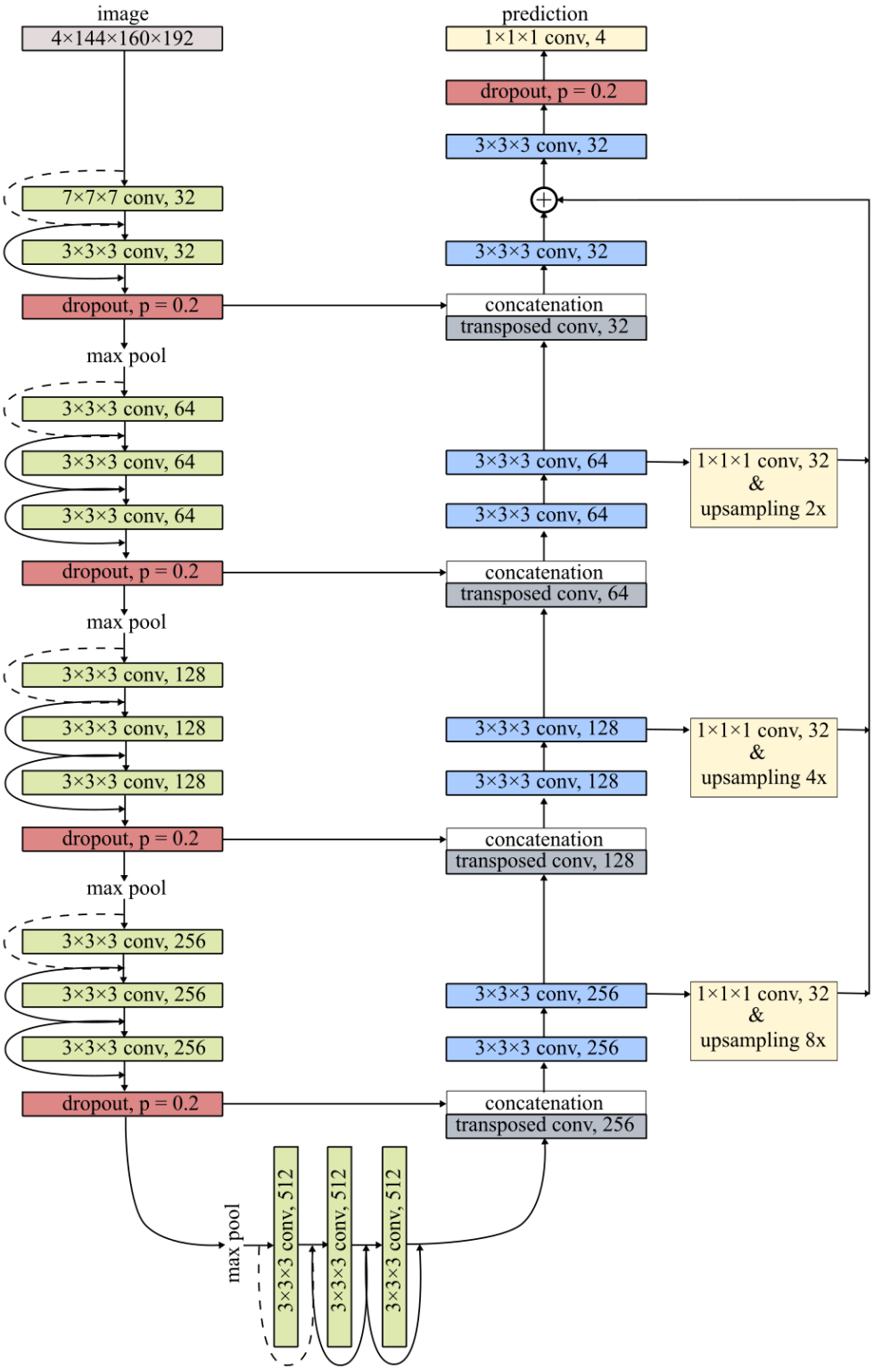


Figure 6.2: Proposed network architecture with SE Normalization.

In both definitions, $y_i = \left[y_i^1, y_i^2, \ldots, y_i^{\mathrm{C}}\right]^\top$ - the one-hot encoded label for the *i-th* voxel, $\hat{y}_i = \left[\hat{y}_i^1, \hat{y}_i^2, \ldots, \hat{y}_i^{\mathrm{C}}\right]^\top$ - predicted probabilities for the *i-th* voxel. N and C are the total numbers of voxels and classes for the given example, respectively. In addition, we apply Laplacian smoothing by adding +1 to the numerator and denominator in the Soft Dice Loss to avoid the zero division in cases where one or several classes are not represented in the training example. The parameter γ in the Focal Loss is set at 2.

The training data in the challenge has labels for three tumor sub-regions, namely the necrotic and non-enhancing tumor core (NCR & NET), the peritumoral edema (ED) and the Gd-enhancing tumor (ET). However, the evaluation is done for the Gd-enhancing tumor (ET), the tumor core (TC), which is comprised of NCR & NET along with ET, and the whole tumor (WT) that combines all provided sub-regions. Hence, during training we optimize the loss directly on these nested tumor sub-regions.

Table 6.1: Our results on the online validation set ($n = 125$). Average values across all patients are provided for each evaluation metric. 'Best Model' corresponds to the best-performing model in the ensemble. The abbreviation 'PP' stands for post-processing.

Metrics	DSC			Sensitivity			HD		
Class	ET	WT	TC	ET	WT	TC	ET	WT	TC
U-Net	0.772	0.899	0.825	0.794	0.896	0.813	5.81	5.97	6.58
Best Model	0.740	0.908	0.862	0.816	0.909	0.854	3.84	4.60	5.34
Ensemble	0.761	0.911	0.863	0.814	0.908	0.850	3.70	4.48	4.82
Ensemble (PP)	0.780	0.911	0.863	0.815	0.908	0.850	3.72	4.48	4.82

6.3.5 Ensembling

To reduce the variance of the model predictions, we build an ensemble of models that are trained on different splits of the train set, and use their average as the ensemble prediction. At each iteration, the model is built on 90%/10% splits of the training set and subsequently evaluated on the online validation set. Having repeated this procedure multiple times, we choose 20 models with the highest performance on the online validation set and combine them into the ensemble. Predictions on the test set are produced by averaging predictions of the individual models and applying a threshold operation with a value equal to 0.5.

Table 6.2: Our results on the test set ($n = 166$). Average values across all patients are provided for each evaluation metric.

Metrics	DSC			Sensitivity			HD		
Class	ET	WT	TC	ET	WT	TC	ET	WT	TC
Ensemble (pp)	0.805	0.887	0.843	0.854	0.909	0.866	15.43	4.54	19.59

6.3.6 Post-processing

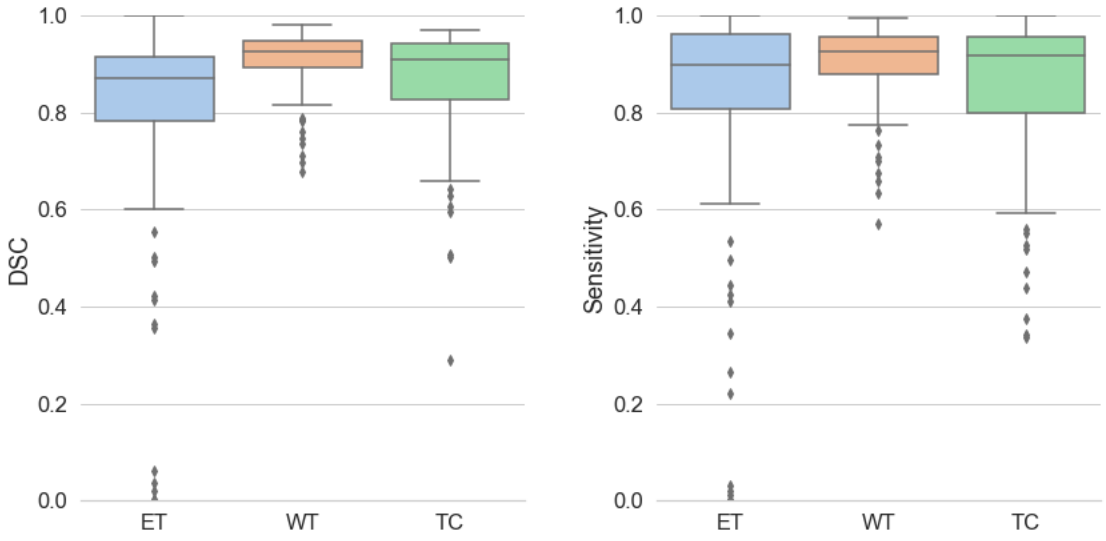
The DSC metric used for the performance evaluation in the challenge is highly sensitive to cases, wherein the model predicts classes that are not presented in the ground-truth (DSC = 0 in such cases). Therefore, a false positive prediction for a single voxel leads to the lowest value of the DSC and might significantly affect the average model performance on the whole evaluation dataset. This primarily refers to patients without ET sub-regions. To address this issue, we add a post-processing step to remove small ET regions from the model outcome, if the ET volume is less than a certain threshold. We set the threshold value at 32 voxels, since it is the smallest ET area among all patients in the training set.

6.4 Results and Discussion

Our results on the online validation set and the test set are summarized in Table 6.1 and Table 6.2, respectively. The final ranking of the best participants is presented in Table 6.3.

On the online validation set with 125 patients without available ground-truth masks, our ensemble of 20 models, fitted on different splits of the training set and with applied post-processing, obtained the DSC of 0.78, 0.911 and 0.863 for the ET, WT and TC glioma sub-regions, respectively. The method applied for post-processing excluded small regions that were incorrectly identified as the ET class that increased the average DSC for this class (from 0.761 to 0.780). Combining multiple models with the same architecture into the ensemble led to marginal improvements in the results for the ET sub-region (DSC increased from 0.740 to 0.761) and the TC sub-region (HD decreased from 5.34 to 4.82) over the single best model. The single model that showed the highest results among others ('Best Model' in Table 6.1) outperformed U-Net in all cases, except for the ET class (DSC of 0.772 vs 0.740).

Figure 6.3 depicts the distribution of the validation set results. Similarly to the segmentation tasks presented in Chapter 4 and Chapter 5, there exists a relatively small set of examples that were poorly segmented. In the BraTS challenge, it mainly



(a) DSC (b) Sensitivity

Figure 6.3: Distributions of the results for the ET, WT and TC glioma sub-regions on the online validation set.

applies to ET sub-regions that can be partially explained by the smaller size of this class compared to other glioma parts. Also, based on the high results for WT, it can be inferred that in some cases ET regions are misidentified by the model as different tumor classes. However, the results for the WT region (DSC 0.911 and sensitivity 0.908) indicate that the model quite rarely recognises glioma sub-regions as the background.

Table 6.3: Final ranking on the test set ($n = 166$). Average values for all metrics are reported for all metrics.

Team	Dice Score			HD			Rank
Class	ET	WT	TC	ET	WT	TC	–
MIC_DKFZ [112]	0.820	0.889	0.851	17.81	8.50	17.34	1
NPU_PITT [116]	0.828	0.888	0.854	13.04	4.53	16.92	2
Radicals [223]	0.816	0.891	0.842	17.79	6.24	19.54	
deepX [234]	0.818	0.883	0.843	13.43	5.22	17.97	3
INSERM (ours)	0.805	0.887	0.843	15.43	4.54	19.59	4

The challenge rules allowed to make a single prediction on the test set (165 patients), which was used to rank all participating teams. In general, our results on the test set in terms of the DSC and sensitivity metrics are consistent with the results obtained in cross-validation (the difference is less than 0.03 for both metrics). However,

there is a high discrepancy between the HD values, which is presumably due to the evaluation procedure. In all cases where the model predicts a class that is absent in the ground-truth, even if it is a single voxel, the scoring system assigns the DSC of 0 and HD of 373 to this class, that might significantly reduce the average values of these metrics. This issue is also evident in the average results of the top-ranking participants (see Table 6.3). However, it could not affect positions in the final leaderboard that was built on the final ranking score (FRS) described in Section 6.2.3.

The team 'MIC_DKFZ' won first prize with nnU-Net framework [114]. Two teams (NPU_PITT and Radicals) tied for second place in the challenge with a statistically insignificant difference between their results (based on permutation testing introduced in Section 6.2.3). Our team (INSERM) finished fourth in the competition, slightly below the leaders, presumably because of the results for ET sub-regions. An example of our predictions on the validation set is shown in Figure 6.4.

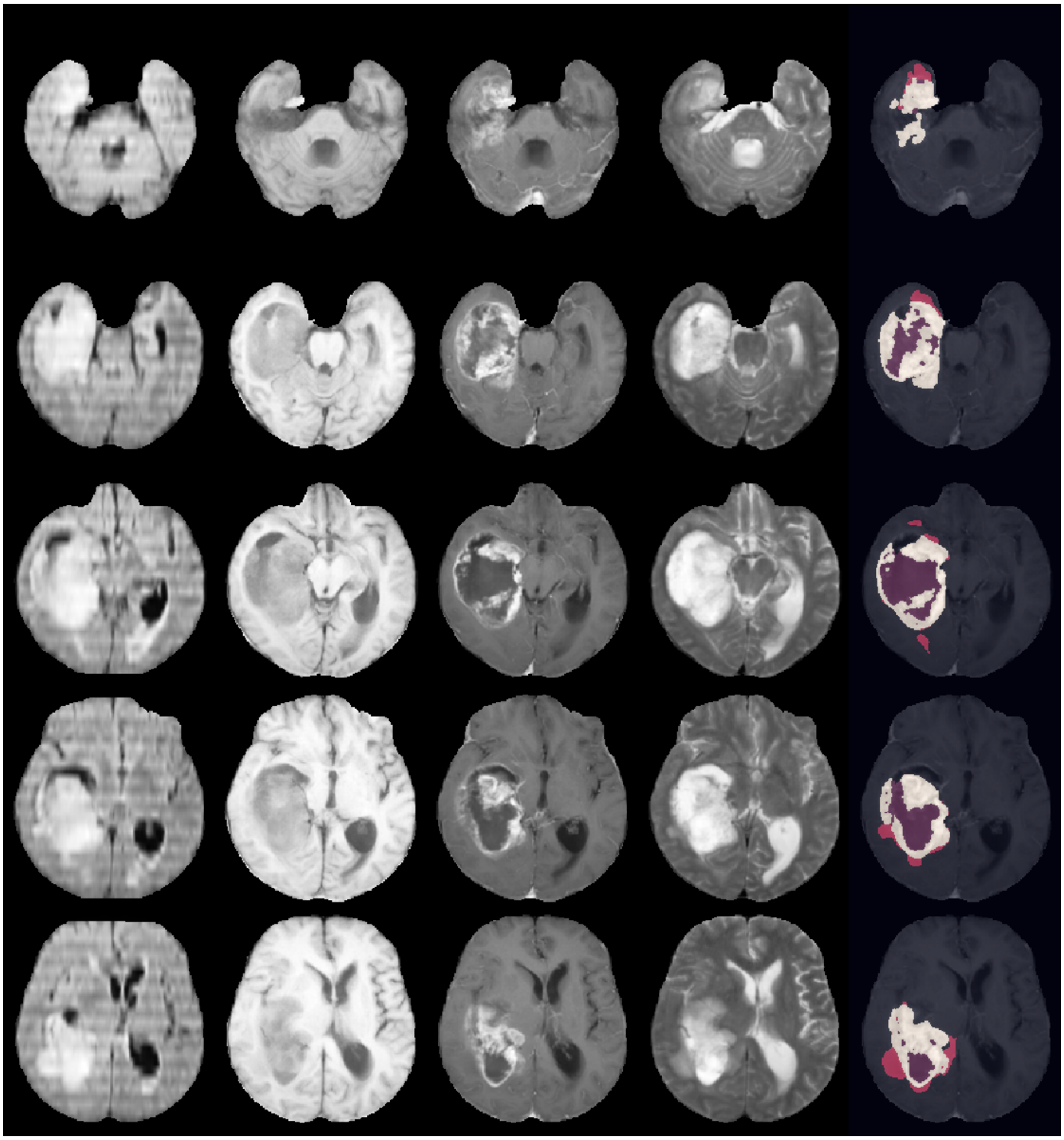


Figure 6.4: Model prediction for the patient (65) from the online validation set. Five axial slices. From left: T2-FLAIR, T1, T1-Gd, and T2. The results for this patient (DSC of 0.892, 0.915, 0.922) are approximately equal to the median values of the online validation set (DSC of 0.872, 0.927, 0.910) for the ET, WT and TC glioma sub-regions, respectively.

Chapter 7

Detection and Segmentation of Lymphoma and Sarcoidosis

Reference

Fully Automated Detection and Segmentation of Hypermetabolic Lesions in Pretherapeutic [18F]FDG PET/CT Images of Lymphoma and Sarcoidosis Patients. Andrei Iantsen, Pierre Lovinfosse, Marta Ferreira, Alexandre Jadoul, Nadia Withofs, Céline Derwael, Anne-Noëlle Frix, Julien Guiot, Dimitris Visvikis, Mathieu Hatt, Roland Hustinx. In:*European Journal of Nuclear Medicine and Molecular Imaging 48 (Suppl 1)* (2021).

Abstract

Aim/Introduction: Automated detection and segmentation of pathological uptakes in [18F]FDG PET images can be useful to derive clinically relevant metrics such as total tumor burden for diagnosis and prognosis purposes. It remains a challenging task given the large possible range of number, location, size and heterogeneity of lesions. Semi-automated delineation such as the use of manually adjusted thresholds applied to visually detected lesions by an expert, remains time-consuming and subjective, and a fully automated approach is thus desirable for improved robustness and reproducibility. The goal of this work was to evaluate the

feasibility of achieving fully automated detection and delineation by training a deep convolutional neural network. **Materials and Methods:** A cohort of 419 patients who underwent pretherapeutic [18F]FDG PET and associated low dose CT scans was retrospectively collected for the purpose of developing and testing the proposed algorithm. The ground-truth for each PET/CT scan was determined semi-automatically by one of four physicians following the same procedure, i.e. standardized uptake value (SUV) threshold of 3, volume > 2 cc and manual correction whenever deemed necessary. The seminal U-Net architecture was applied "of the shelf" to develop the model on 397 patients and evaluate it on 22 test patients. In addition, the test subset was annotated by all experts independently to evaluate inter-observer variability. Dice similarity coefficient (DSC), Sensitivity (SE) and Positive Predictive Value (PPV) computed on a patient basis (i.e., all lesions considered together) were used for the first stage evaluation. A lesion by lesion analysis was then performed applying different detection criteria. An ablation study was carried out to identify main factors affecting segmentation results. **Results:** The model obtained good average accuracy for all metrics on the patient basis (DSC $= 0.84 \pm 0.16$) with SE (0.84 ± 0.21) and PPV (0.90 ± 0.12). On the lesion basis, the performance varied (DSC between 0.61 and 0.77; SE 0.60 - 0.75; PPV 0.66 - 0.83) depending on the chosen detection criteria. The analysis of the inter-observer variability demonstrated insignificant differences between the ground-truth annotations of all experts (e.g., the patient-wise DSC $= 0.96 \pm 0.15$) and ensured the reproducibility of the procedure for establishing the ground-truth. Visual inspection confirmed the relevance of the model predictions and revealed the limitations inherent to the evaluation method. **Conclusion:** The proposed approach achieved good overall results and might provide a robust and accurate fully automated solution for future works investigating the clinical prognostic and predictive value of metrics derived from these segmentation masks.

A part of this chapter is based on the materials presented at the 34th Annual Congress of the European Association of Nuclear Medicine – EANM'21 Virtual.

7.1 Introduction

Lymphoma is a type of lymphoproliferative disorder, characterized by malignant transformations in lymphocytes that can spread to various parts of the body, including lymph nodes, spleen, bone marrow and other organs, and subsequently form tumors. Two broad groups of this disease, Hodgkin lymphoma (HL) and non-Hodgkin lymphoma (NHL), amount to 83 087 and 544 352 new cases (23 376 and 259 793 deaths), respectively, worldwide in 2020 [206].

Positron emission tomography/computed tomography (PET/CT) imaging is an integral instrument in oncology, generally used for diagnosis, baseline staging, treatment planning and response assessment. Many lymphomas have characteristic morphological features that appear in CT and are suitable for detection of sites of disease. PET findings, on the other hand, provide functional information with a lack of anatomic landmarks by indicating the overall level of metabolic activity of lymphoma. In case of ^{18}F–fluorodeoxyglucose (FDG) radiotracer, uptake rate often correlates positively with tumor aggressiveness [17, 54, 62, 126, 195] and some clinical prognostic factors, such as serum lactate dehydrogenase (LDH) level [43, 193].

The standardized uptake value (SUV) and its derivatives (e.g., SUVmax and SUVmean), metabolic tumor volume (MTV, a measure of the metabolically active, i.e., exceeding a certain SUV threshold, tumor volume), and total lesion glycolysis (TLG, averaged SUV multiplied by MTV) are quintessential descriptors that can be extracted from PET/CT images in order to build diagnostic, prognostic and predictive tools [42, 43, 190]. The use of more sophisticated features for quantification of tumor phenotypes is investigated in the field of radiomics [64, 85, 157, 221]. For instance, the seminal article by Aerts et al. [1] revealed that radiomic features could capture intratumor heterogeneity and, furthermore, are associated with underlying gene-expression patterns. Although it was later shown that the specific signature developed in that work was only a surrogate of the tumor volume [225].

A number of recent articles have described promising results of radiomics analysis carried out in patients with various types of lymphoma. Zhou et al. [246], for instance, relied on radiomic features, extracted from PET/CT images, to predict overall and progression-free survival (OS and PFS) in case of diffuse large B-cell lymphoma (DLBCL). Lue et al. [149] used clinical and PET/CT radiomic features to evaluate response to therapy, PFS and OS in patients with HL. Milgrom et al. [157] identified five radiomic features, including MTV and TLG, that were highly predictive of primary refractory status in a cohort of patients with early-stage HL. In terms of

an area under the curve (AUC), their model with additional features received significantly more accurate predictions (0.95), when compared to MTV (0.78), TLG (0.78) or SUVmax (0.65) alone. Relying on the radiomics approach, Lippi et al. [140] differentiated between four lymphoma subtypes on a dataset comprised of 60 patients, and demonstrated a convincing predictive performance with recall of 0.97 and precision of 0.94. In addition, radiomics could be applied to accurately distinguish lymphoma from another types of cancer in PET/CT images [125, 169, 170].

In radiomics workflow, features are to be extracted from predetermined tumor volumes, which are commonly defined manually or semi-automatically by one or many clinical experts. Semi-automatic segmentation of lymphoma lesions in PET/CT has traditionally been performed by applying different thresholding techniques within volumes of interest (VOIs), previously detected in the entire scan [107, 153]. The fundamental principle of such methods is to treat all regions with uptake above a certain threshold as tumors. The threshold value can be fixed (e.g., SUV of 2.5 or 4 [15, 63, 128]) or adaptively chosen, based on the uptake distribution in the whole scan (e.g., 25% or 41% of SUVmax [30, 31, 155]) or some reference region (e.g., mean liver uptake [220]). Since user interaction is an essential part, these annotation procedures are not only time-consuming but also acutely sensitive to observer-related variability. As a consequence, the radiomic features are significantly affected by the choice of image segmentation methods, applied to distinguish tumors from normal tissues, and the observers themselves [84, 123, 173, 182, 236]. In lymphoma, segmentation is especially challenging, given the large possible range of number, location, size and heterogeneity of lesions in the body. A fully automated approach to lymphoma segmentation is thus strictly necessary for improved robustness and reproducibility.

Convolutional neural networks (CNNs) have consistently achieved state-of-the-art results in most visual recognition tasks. A lack of a unique, sufficiently large dataset, like ImageNet [45], in the medical imaging domain has resulted in a myriad of CNN architectures presented in the literature (see Chapter 3). Nonetheless, in the case of medical image segmentation, the vast majority of the architectures stem from U-Net [34, 183], an encoder-decoder network with skip connections. Although it is commonly reported that alternative architectures have superior performance in comparison with U-Net, recent publications imply that the benefits of these models are likely to be restricted to datasets on which they were trained and tested [113, 141, 213, 214]. Moreover, other components of the general pipeline, e.g., data preprocessing and augmentation, training procedure, etc., can have a much greater impact on the outcome than tricky modifications in the architecture design. As shown by Isensee

et al. [113], the orthodox U-Net (with adjustments only in the number of layers and filters) surpasses most existing, highly specialized networks on 23 public datasets used in international biomedical segmentation competitions, if the other pipeline components are properly selected. Hence, only the U-Net architecture is considered in this chapter.

Although PET/CT imaging plays a key role in lymphoma evaluation, it is also fraught with potential pitfalls, mainly associated with its limited specificity. Lymphoma lesions exhibit variable FDG uptake that can be difficult to detect, particularly in the presence of elevated physiologic uptakes that potentially result in false-positive findings. Normally, physiologic uptake is observed in the brain, heart, liver, spleen, gastrointestinal tract, urinary collecting system (including the bladder) and bone marrow [196]. Nonetheless, intense FDG uptake can also be a sign of tissue inflammation caused by another disorder [53, 145, 189]. A typical example is sarcoidosis, which is a systemic disease of unknown etiology, characterized by the presence of (benign) granulomatous lesions in various organs, primarily the lungs and the lymphatic system [28, 58]. Moreover, both diseases can coexist, with sarcoidosis usually preceding lymphoma, and a biopsy is necessary to obtain histological evidence for the presence of malignancy [27, 143, 171, 175]. Since biopsy sites must be localized before treatment planning, segmentation of both lymphoma and sarcoidosis simultaneously is considered in this chapter.

The performance of automated segmentation is commonly evaluated with volume- and distance-based metrics, computed on the ground-truth and predicted binary segmentation mask of the entire scan (see Section 6.2.3). In lymphoma, however, multiple lesions of different sizes can be situated in various parts of the body at a relatively large distance from each other. As a result, these metrics can be misleading when measuring segmentation performance, especially if some lesions are missed in the outcome. Moreover, detection of all individual lesions, even with coarse contours, is more important for therapy planning to reduce a risk of relapsed and refractory lymphoma in future. Therefore, two complementary groups of metrics, based on the Dice similarity coefficient (DSC), sensitivity (SE) and positive predictive value (PPV), are used in this chapter for performance evaluation. More specifically, segmentation is considered *voxel-wise*, meaning that the overlap between the ground-truth and prediction is estimated for the total tumor volume composed of individual lesions. Detection, on the other hand, is estimated *lesion-wise* and quantifies a fraction of lesions that are correctly detected according to some criteria (see details in Section 7.2.4).

The purpose of this chapter is thus to evaluate the feasibility of achieving fully

automated detection and segmentation of both lymphoma and sarcoidosis lesions in pretherapeutic ^{18}F–FDG PET/CT images by applying the U-Net model.

7.2 Materials and Methods

The study has been approved by the Ethics Committee of the University Hospital of Liège. The need for written informed consent was waived due to the retrospective and non-interventional nature of the study.

7.2.1 Data Description

A dataset consisting of 419 patients with Hodgkin lymphoma (HL, $n = 140$), diffuse large B-cell lymphoma (DLBCL, $n = 111$) or sarcoidosis ($n = 168$) was provided by the University Hospital of Liège. All cases of lymphoma and some of the diagnoses of sacroidosis were confirmed with biopsy, whereas the remaining diagnoses were based on clinical evidence and follow-up.

All ^{18}F–FDG PET/CT scans were acquired between April 2010 and February 2020, using two PET/CT scanners, namely a GEMINI TF Big Bore and a GEMINI TF 16 (Philips Medical Systems, Cleveland, OH, USA), according to the European Association of Nuclear Medicine Research Limited (EARL) guidelines. The PET images were reconstructed with a voxel size of $4 \times 4 \times 4$ mm^3 using a blob-based iterative time-of-flight reconstruction algorithm (BLOB-OS-TF) that included CT-based attenuation and scatter corrections, without post-reconstruction smoothing.

To generate ground-truth (gold standard) segmentation masks, the entire dataset was divided into four non-overlapping subsets, and each was independently annotated by one of four nuclear medicine physicians with 3, 6, 10 and 15 years of clinical experience. Segmentation was performed in a semi-automatic manner and each observer followed a standardized procedure in order to reduce variability in the ground-truth definition. First, fixed thresholding with SUV of 3 was applied to localize regions of high metabolic activity in PET. Next, all volumes smaller than 2 cubic centimeters (cc) were excluded, because their uptake values could be significantly biased due to the partial volume effect [81, 168, 201]. Then, sites of physiological uptake in the brain, liver, kidney, and other parts of the body were manually removed. After, the remaining volumes were treated as pathological lesions and their contours could be manually adjusted if necessary. Last, all lesions were marked using natural numbers and the remaining area, i.e., the background, was labeled with zeros to obtain

the corresponding ground-truth mask. Each physician was unaware of any clinical information and diagnosis, performing segmentation solely on the PET/CT images.

All data was split into training, validation and test sets consisting of 357, 40 and 22 patients, respectively. To alleviate a possible discrepancy between data distributions in the folds, all examples were sampled using the average lesion volume for stratification.

7.2.2 Network Architecture

The famous U-Net architecture [34, 183] was applied "off the shelf" and only basic hyperparameters, namely a layer width (i.e., a number of channels), a number of layers and a number of stages, were fine-tuned on the validation set.

U-Net is an encoder-decoder network that includes skip connections to copy some feature maps from the encoder to the decoder. Usually, this architecture is formed by a set of convolutional blocks, and each block is composed of a convolutional layer, a normalization layer and a nonlinear transformation applied element-wise. Throughout this chapter, all convolutional layers have the kernel size of $3 \times 3 \times 3$ and are applied with padding to keep the spatial size of the input unchanged. After each convolution, the feature maps are normalized using instance normalization, and followed by the ReLU activation to capture nonlinear patterns in data.

The model has a number of stages, characterized by the spatial size of the feature maps. At each stage in the encoder, two convolutional blocks are applied to generate the stage output that is stored to be subsequently transferred to the decoder through skip connections. First convolutional blocks, placed right after the model input, have the layer width of 32.

Transition to the next stage in the encoder is performed with max pooling that halves each side of the feature maps. At the following stage, the number of feature maps (i.e., the layer width) is doubled to compensate for the information loss caused by downsampling. In the decoder, on the contrary, the number of feature maps is first halved by a pointwise convolution (i.e., it uses a $1 \times 1 \times 1$ kernel) with nonlinearity, and nearest neighbor interpolation is then applied to double each side of the corresponding feature maps. The model used in this chapter consists of four stages in total, i.e., max pooling is applied three times.

The final layer is a pointwise convolution with the sigmoid activation to generate the probability map, i.e., soft labels. The probability map is then converted into a segmentation mask by applying thresholding with a value of 0.5 to produce binary,

i.e., hard, labels for all individual voxels.

7.2.3 Training and Inference

Training is performed on $128 \times 128 \times 64$ patches that are randomly extracted from the training images and combined into batches such that each is composed of two patches. A training budget is set to 100 epochs and each epoch is equivalent to 1 000 batches. During training, a learning rate is adjusted using a cosine annealing learning rate schedule [146]: starting from an initial value of 10^{-4}, it is gradually reduced to a minimum value of 10^{-7} within a cycle of 10 epochs. After each cycle, a warm restart is performed by setting the learning rate to the initial value. This procedure enables the model to converge to multiple local minima in the course of training, which improves generalization [102, 146, 199]. Training is performed with Adam, an algorithm for first-order gradient-based optimization of stochastic objective functions [122].

The unweighted sum of the Soft Dice Loss [158] and the Focal Loss [139] is employed as the objective function. Based on [158], the Soft Dice Loss, a differentiable surrogate for the DSC metric, can be written for one data example as

$$L_{Dice}(y, \hat{y}) = 1 - \frac{2\sum_i^{\mathrm{N}} y_i \hat{y}_i + 1}{\sum_i^{\mathrm{N}} y_i^2 + \sum_i^{\mathrm{N}} \hat{y}_i^2 + 1}\,. \tag{7.1}$$

The Focal Loss is defined as

$$L_{Focal}(y, \hat{y}) = -\frac{1}{\mathrm{N}} \sum_i^{\mathrm{N}} y_i (1 - \hat{y}_i)^{\gamma} \log(\hat{y}_i)\,. \tag{7.2}$$

In both definitions, $y_i \in \{0, 1\}$ - the label for the *i-th* voxel, $\hat{y}_i \in [0, 1]$ - the predicted probability for the *i-th* voxel, and N - the total numbers of voxels. Also, we add $+1$ to the numerator and denominator in the Soft Dice Loss to avoid a division by zero in cases where the lesion class is not present in training patches. The parameter γ in the Focal Loss is set to 2.

During inference, the trained model makes predictions on the whole PET/CT scan using a sliding window approach with a $64 \times 64 \times 32$ stride, i.e., adjacent patches have a 50% overlap, to avoid edge artifacts. Then, the output is binarized with a 0.5 threshold to obtain the segmentation mask.

Generally speaking, object detection entails both marking an object with a certain label and localizing this object with an individual bounding box [61, 138]. However, direct prediction of bounding boxes on 3D medical images is often impracticable due

to the limited amount of data available for training. Therefore, it is common practice to first mark all objects with a single class label, i.e., perform binary segmentation, and then re-mark them with individual instance labels [183, 224]. The latter can be addressed as a connected-component labeling problem [93, 197, 229]. Thus, all neighboring voxels in the segmentation mask are considered as part of the same lesion and re-marked with the unique label.

Last, all predicted lesions smaller than 2 cc are excluded during post-processing.

7.2.4 Evaluation Metrics

Given a binary mask predicted by the model, the segmentation quality is assessed using the Dice similarity coefficient (DSC), sensitivity (SE) and positive predictive value (PPV) computed on a *voxel basis* as follows:

$$\text{DSC} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}}, \quad \text{SE} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{PPV} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{7.3}$$

where TP is a number of true positive voxels, FP is a number of false positive voxels, and FN corresponds to false negative voxels.

Consider a ground-truth lesion as a true positive, i.e., correctly detected, if there is a lesion predicted by the model such that the value of DSC computed on binary masks corresponding to these lesions is greater than or equal to a certain threshold t. If this value is less than t, the ground-truth lesion is treated as a false negative. Thus, the evaluation metrics in Equations 7.3 can be rewritten on a *lesion basis*:

$$\text{DSC}_t = \frac{2\text{TP}_t}{2\text{TP}_t + \text{FP}_t + \text{FN}_t}, \quad \text{SE}_t = \frac{\text{TP}_t}{\text{TP}_t + \text{FN}_t}, \quad \text{PPV}_t = \frac{\text{TP}_t}{\text{TP}_t + \text{FP}_t}, \tag{7.4}$$

where TP_t is a number of true positive lesions (depending on the threshold t), FN_t is a number of false negative lesions, and a number of false positive lesions can be computed as $\text{FP}_t = \text{P} - \text{TP}_t$, where P - a total number of predicted lesions.

Thus, the segmentation quality of an individual lesion required in the detection task can be adjusted by varying the threshold value t. In this chapter, detection was assessed with respect to three different values of t: 0.25, 0.5 and 0.75.

7.3 Results and Discussion

7.3.1 Interobserver Variability

As each PET/CT scan was randomly assigned to one of four nuclear medicine physicians with different levels of clinical experience for ground-truth annotation, evaluation of interobserver variability was carried out prior to model development. For this purpose, all 22 scans from the test fold were independently segmented by each observer and their results were then compared pairwise using the segmentation metrics. As a results, the average DSC across all possible pairs of the observers was 0.956 ± 0.154, with SE of 0.970 ± 0.135 and PPV of 0.964 ± 0.139, indicating a high level of agreement in the ground-truth definition and reproducibility of the established procedure.

7.3.2 Segmentation

Using the validation set, a number of different model configurations, i.e., hyperparameter choices, were assessed with the DSC metric. The best performance was obtained by the model with 32 channels, 4 stages and 2 layers at each stage. Its wider and deeper counterparts either had no effect on the performance, while being more computationally expensive, or even worsen the results.

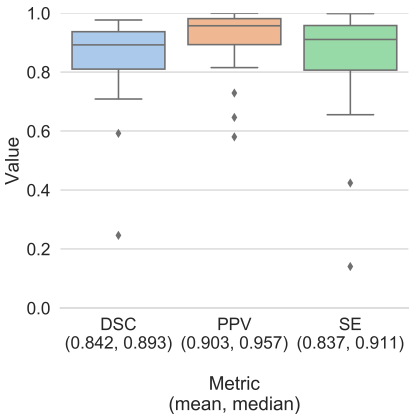
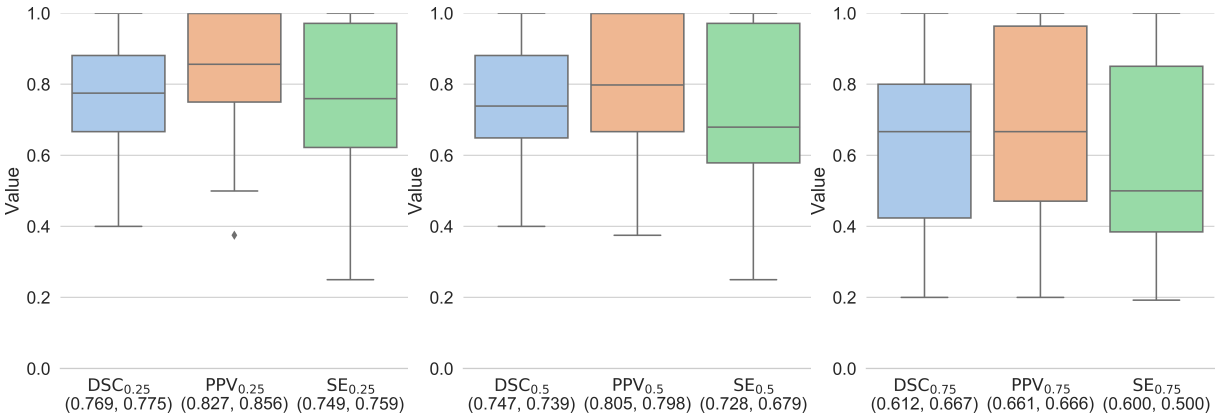


Figure 7.1: Segmentation results on the test set.

The best model had the average DSC values of 0.835 and 0.842 on the validation and test sets, respectively. In addition, it demonstrated similar results on the training set as well (DSC = 0.837), showing great consistency in segmentation performance on all folds. Given the average values of PPV significantly larger than SE (0.903 vs 0.837), the model had a tendency to underestimate the total lesion volume. In a few cases, the model yielded poor predictions, which significantly reduced the average of each metric (see Figure 7.1).

The distributions of the segmentation results on the test set for the target classes, i.e. HL, DLBCL and sarcoidosis, are shown in Figure 7.2. The best outcome was obtained for patients with DLBCL (DSC = 0.909), while segmentation of HL lesions proved to be most challenging among the considered classes (DSC = 0.744). However,



(a) $t = 0.25$ (b) $t = 0.5$ (c) $t = 0.75$

Figure 7.3: Distributions of the detection results on the test set for different thresholds.

these findings are likely to be biased in the light of the relatively limited size of the test set. The predicted binary segmentation masks were transformed into the lesion-labeled masks by applying connected-component labeling and the post-processing step that excluded volumes less than 2 cc.

7.3.3 Detection

The detection results for the different threshold values ($t = 0.25, 0.5, 0.75$) are summarized in Figure 7.3. As the threshold increases, stricter requirements are imposed on the model. Therefore, the highest values of all metrics related to the lowest threshold, $t = 0.25$, (Figure 7.3a) when it was sufficient for the model to correctly segment just a small fraction of the lesion. In this case, the average $\text{DSC}_{0.25}$ computed on the test set was 0.769, with $\text{PPV}_{0.25}$ significantly higher than $\text{SE}_{0.25}$ (0.827 vs 0.749, respectively), indicating that some lesion were missed by the model.

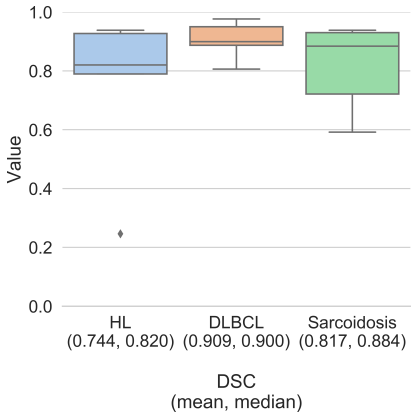


Figure 7.2: Segmentation results for each class on the test set.

If the highest threshold was set, the mean values of all metrics substantially decreased and at the same time a noticeable increase in their variance occurred (Figure 7.3c). For example, by increasing the threshold value from 0.5 to 0.75, each detection metric declined by about 0.13. On the other hand, using the threshold of

0.5 instead of 0.25 resulted only in a marginal drop (by 0.022 at worst) in the average values of all metrics. Hence, the threshold of 0.5 can be considered as a good trade-off, providing both acceptable segmentation of individual lesions and high detection performance (Figure 7.3b).

Examples of the model predictions for different lesion types are shown in Figure 7.4. Each individual lesion was marked with a unique, natural number corresponding to some color. When calculating the detection metrics, the lesion numbering had no effect on the outcome.

In the first example (Figure 7.4, first row), the model showed significantly better results in terms of segmentation (DSC = 0.789, PPV = 0.992, SE = 0.655) than detection ($DSC_{0.5}$ = 0.592, $PPV_{0.5}$ = 0.615, $SE_{0.5}$ = 0.571). This was primarily due to the presence of a large number of small lesions, and each of them could either be completely missed by the model or marked as several separate lesions. In such cases, accurate detection is unlikely to be feasible, as the boundaries between adjacent lesions are adversely influenced by the partial volume effect. In addition, many ground volumes are highly non-convex, which makes segmentation metrics more reliable for performance assessment. In the second case (Figure 7.4, second row), a patient with DLBCL, the trained model showed convincing results for both groups of metrics, namely DSC = 0.886, PPV = 0.997, SE = 0.797, and $DSC_{0.5}$ = 0.933, $PPV_{0.5}$ = 0.999, $SE_{0.5}$ = 0.874. Unlike the first case, this patient did not have clusters of closely spaced lesions, which had a favourable impact on detection performance. The last example corresponds to a patient with sarcoidosis that has a massive group of lesions mainly located in the lungs (Figure 7.4, last row). The model prediction for this patient was accurate in terms of segmentation (DSC = 0.884, PPV = 0.893, SE = 0.875), whereas the quality of detection was substantially deteriorated ($DSC_{0.5}$ = 0.750, $PPV_{0.5}$ = 0.749, $SE_{0.5}$ = 0.749) due to a few isolated lesions of small volume, incorrectly labeled by the model.

7.4 Conclusion

The described model obtained good average accuracy for all metrics on the voxel basis (DSC = 0.842 ± 0.163, PPV = 0.903 ± 0.117, SE = 0.836 ± 0.206). On the lesion basis, the performance varied (DSC between 0.612 and 0.769; SE 0.600 – 0.749; PPV 0.661 – 0.827) depending on the chosen detection criteria. The analysis of the inter-observer variability demonstrated insignificant differences between the ground-truth annotations of all clinical experts (e.g., the voxel-wise DSC = 0.956 ± 0.154)

and ensured the reproducibility of the procedure for establishing the ground-truth. Visual inspection confirmed the relevance of the model predictions and revealed the limitations inherent to the evaluation metrics.

The presented approach achieved good overall results and might provide a robust and accurate fully automated solution for future works investigating the clinical prognostic and predictive value of metrics derived from these segmentation masks. In addition, the obtained results can serve as a starting point for future studies aimed at differentiating lesion types based directly on medical images, i.e. without performing a biopsy.

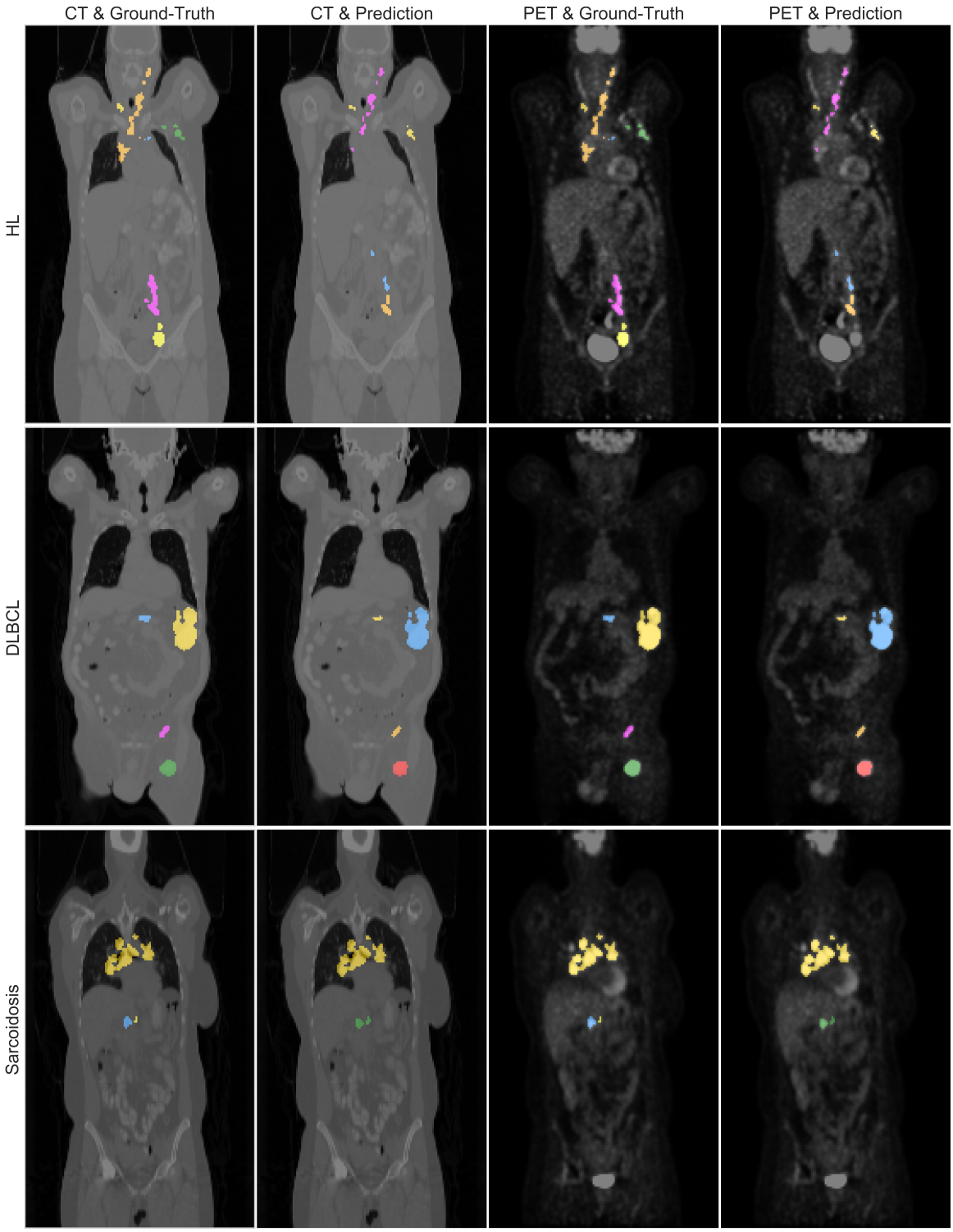


Figure 7.4: Examples of the predictions on the test set.

Chapter 8

Conclusions and Perspectives

In this thesis, four practical cases of applying CNN-based methods to tumor segmentation in medical images of various modalities, including multimodal imaging, were examined. For two of them, results were obtained in public contests with multiple participants, whose solutions provided additional benchmarks for performance assessment.

It is already evident from numerous studies and the cases considered in this thesis that existing AI methods, primarily CNNs, are capable of providing impressive results in medical image segmentation tasks. Moreover, such approaches can outperform human experts by making more accurate predictions, as demonstrated in Chapter 5. In contrast to humans, these models are free from intra-observer variability and can run much faster as well. For example, inference time of the model in Chapter 7 is just about 6 sec per patient on a GPU. Nonetheless, the potential integration of such models in clinical workflows remains highly questionable for most practitioners.

Major concerns are directly related to the reliability of these methods in different circumstances. In the medical imaging domain, there is not a diverse, sufficiently large and annotated dataset, like ImageNet, commonly employed by the research community to develop new models and make fair comparisons between them. So far, important findings presented in the literature have been obtained for relatively small, often private cohorts of patients with a certain type of disorder. As a result, there exists an excessive amount of sophisticated CNN architectures fine-tuned, probably even overfitted, for a particular task and/or dataset. However, as shown by Isensee et al. [113], other components of the general pipeline, e.g., data preprocessing, augmentation, training procedure, etc., can have a much greater impact on the outcome than tricky modifications in the architecture design. Therefore, collective efforts of the research community in future should be more focused on constructing a diverse,

large-scale dataset rather than developing yet another model architecture with dubious generalization. Until then, only certain aspects of the reliability of these methods can be investigated relying on available datasets.

Typically, datasets presented in the literature on medical image segmentation relate to a specific disease, such as a particular type of cancer, which means that models are trained to recognize a limited range of pathologies. In practice, however, it is highly desirable to have a multi-disease model capable of accurately detecting various types of abnormalities simultaneously. One such case was considered in Chapter 7, wherein a single CNN was developed to detect and delineate both lymphoma and sarcoidosis lesions at once. The presented model achieved good overall results for all considered disorders, supporting the potential use of CNNs in the multi-disease setting.

The generalization performance of automated segmentation methods can be seriously hindered by the existing diversity in imaging equipment and protocols used in different medical centers. Hence, the equipment neutrality is another critical requirement for the automated methods that should demonstrate the robust performance, when analyzing images in a multi-center context. Results obtained in Chapters 4,5 with the use of specialized cross-validation allow to conclude that designed models are able to provide similar performance on images from different institutions, and center-related variability can be significantly reduced by applying relatively simple data augmentation techniques. For cervical cancer segmentation in PET imaging (Chapter 4), the analysis was carried out using ground-truth annotations generated by a semi-automated algorithm in order to decrease observer-related variability and obtain more reliable estimates of the generalization performance. A new computational unit, referred to as SE Norm, was introduced in Chapter 5 to address the task of head and neck primary tumor segmentation in PET/CT scans in the context of the MICCAI 2020 HECKTOR challenge. The U-Net architecture with residual blocks and SE Norm units made accurate and robust predictions without any center-specific data standardization. The same model, except for a few minor modifications, was also employed in the MICCAI 2020 BraTS competition to delineate different glioma subregions in multisequence MRI. The described approaches obtained highly competitive results in both competitions, essentially without any task-specific adjustments.

Another significant limitation of AI segmentation methods, often neglected by the research community, is the bias towards positive disease detection. The models are typically trained solely on patients with pathology, which results in a high sensitivity but a limited specificity. In imaging tests, it could lead to false positives and/or

potential overdiagnosis in healthy patients. This problem can be addressed in future studies by including disease-free cases in the development process.

The other problem is related with the fact that modern CNNs commonly rely on graphics processing units (GPUs) for accelerated computations, which is a significant technical bottleneck for most medical centers. Thus, it is necessary to investigate diverse strategies for relaxing hardware requirements and limiting the execution time needed for model inference. Pruning, for example, can be used in an attempt to reduce the number of layers without a serious deterioration in the model performance. Also, quantization techniques can be investigated as an alternative to pruning in order to decrease the size of the model weights and accelerate execution.

Appendix A

Plots and Tables

Table A.1: Kolmogorov-Smirnov and Wilcoxon signed-rank tests to compare results of the proposed model and U-Net. Both tests are two-sided and applied to each evaluation metric. Test statistics (T) and corresponding P-values (P) are present in columns. Asterisks indicate statistically significant results with the significance level $\alpha = 0.05$.

Center	Kolmogorov-Smirnov Test						Wilcoxon Signed-Rank Test					
	DSC		Precision		Recall		DSC		Precision		Recall	
	T	P	T	P	T	P	T	P	T	P	T	P
Brest	0.26	.02*	0.25	.03*	0.28	.01*	499	$< .001$*	716	$< .001$*	294	$< .001$*
Nantes	0.30	.24	0.17	.89	0.52	$< .001$*	74	.05	97	.21	11	$< .001$*
Montreal	0.23	.50	0.27	.31	0.27	.31	143	.41	103	.07	116	.13
Barcelona	0.29	.26	0.17	.90	0.25	.45	75	.03*	129	.55	88	.08
Liège	0.07	.99	0.11	.64	0.17	.16	1932	.64	1696	.16	900	$< .001$*

Table A.2: Average results of the proposed model for different *volume* decile groups. The *i-th* decile group corresponds to patients with the tumor *volume* between d_{i-1} and d_i, where d_i - the *i-th* empirical decile of the tumor *volume* distribution.

Metric	Volume Decile Group									
	1	2	3	4	5	6	7	8	9	10
DSC	0.56 ± 0.22	0.71 ± 0.1	0.76 ± 0.19	0.81 ± 0.1	0.83 ± 0.09	0.84 ± 0.08	0.85 ± 0.07	0.84 ± 0.05	0.85 ± 0.07	0.83 ± 0.12
Precision	0.44 ± 0.23	0.59 ± 0.13	0.69 ± 0.19	0.78 ± 0.14	0.80 ± 0.15	0.81 ± 0.13	0.81 ± 0.12	0.83 ± 0.11	0.82 ± 0.12	0.90 ± 0.10
Recall	0.94 ± 0.06	0.93 ± 0.06	0.89 ± 0.2	0.87 ± 0.14	0.91 ± 0.12	0.91 ± 0.11	0.91 ± 0.08	0.88 ± 0.11	0.91 ± 0.09	0.81 ± 0.18

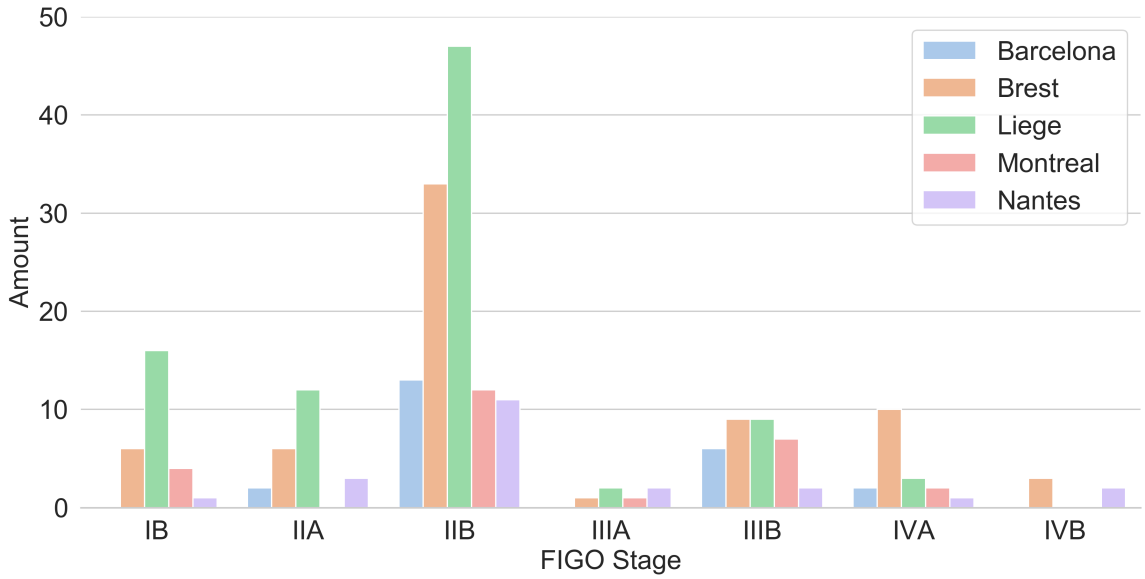
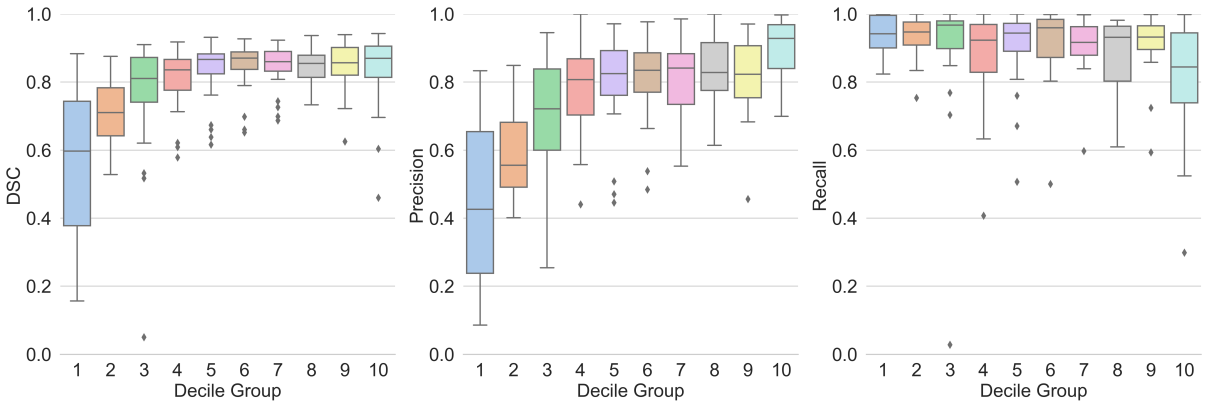


Figure A.1: Distributions of the patients with different FIGO stages in each center.

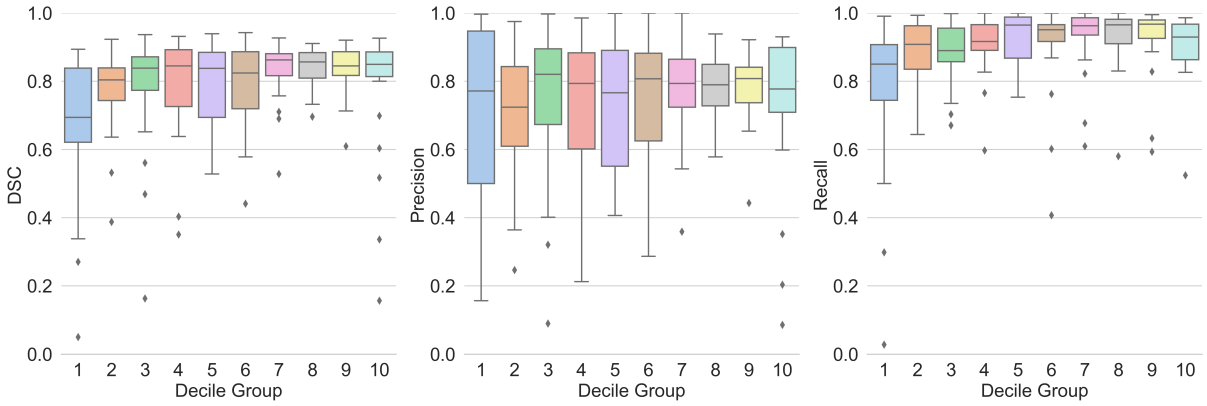
Table A.3: Average results of the proposed model for different *contrast* decile groups. The *i-th* decile group corresponds to patients with the tumor *contrast* between d_{i-1} and d_i, where d_i - the *i-th* empirical decile of the tumor *contrast* distribution. The tumor contrast is defined as a ratio of the average tumor intensity to the average intensity of the body region.

Metric	Contrast Decile Group									
	1	2	3	4	5	6	7	8	9	10
DSC	0.67 ± 0.21	0.77 ± 0.12	0.78 ± 0.18	0.78 ± 0.15	0.79 ± 0.13	0.79 ± 0.13	0.83 ± 0.09	0.84 ± 0.06	0.84 ± 0.07	0.78 ± 0.19
Precision	0.70 ± 0.27	0.71 ± 0.20	0.75 ± 0.24	0.73 ± 0.22	0.73 ± 0.20	0.75 ± 0.20	0.78 ± 0.14	0.78 ± 0.09	0.78 ± 0.10	0.74 ± 0.23
Recall	0.77 ± 0.23	0.89 ± 0.09	0.88 ± 0.09	0.91 ± 0.09	0.92 ± 0.09	0.90 ± 0.14	0.93 ± 0.10	0.93 ± 0.09	0.92 ± 0.11	0.90 ± 0.10



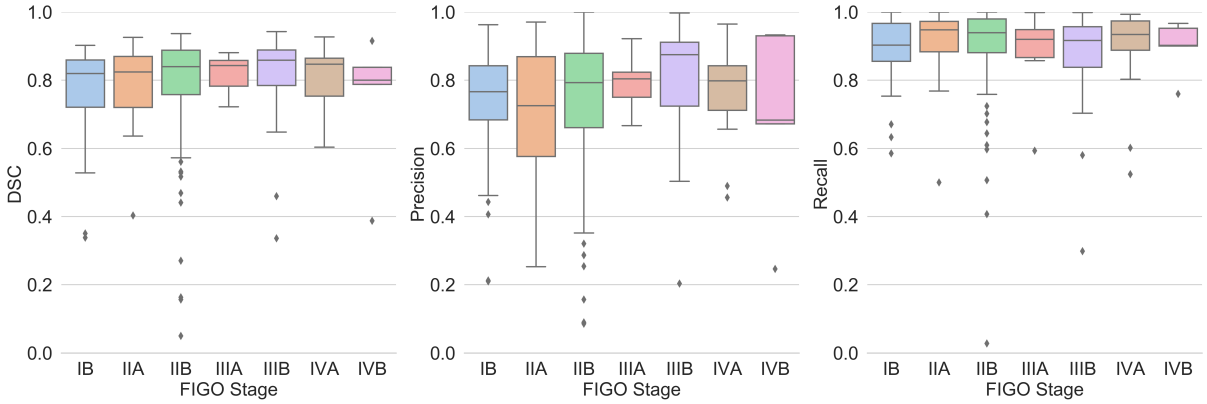
(a) DSC (b) Precision (c) Recall

Figure A.2: Results of the proposed model for different *volume* decile groups. The *i-th* decile group corresponds to patients with the tumor *volume* between d_{i-1} and d_i, where d_i - the *i-th* empirical decile of the tumor *volume* distribution.



(a) DSC (b) Precision (c) Recall

Figure A.3: Results of the proposed model for different *contrast* decile groups. The *i-th* decile group corresponds to patients with the tumor *contrast* between d_{i-1} and d_i, where d_i - the *i-th* empirical decile of the tumor *contrast* distribution. The tumor contrast is defined as a ratio of the average tumor intensity to the average intensity of the body region.



(a) DSC (b) Precision (c) Recall

Figure A.4: Results of the proposed model for different FIGO stages.

Table A.4: Average results of the proposed model for different FIGO stages.

Metric	FIGO Stage						
	IB (n=27)	IIA (n=23)	IIB (n=120)	IIIA (n=6)	IIIB (n=33)	IVA (n=18)	IVB (n=5)
DSC	0.76 ± 0.15	0.78 ± 0.12	0.79 ± 0.16	0.82 ± 0.06	0.81 ± 0.14	0.81 ± 0.10	0.75 ± 0.21
Precision	0.71 ± 0.20	0.72 ± 0.19	0.74 ± 0.21	0.79 ± 0.09	0.80 ± 0.18	0.77 ± 0.14	0.69 ± 0.28
Recall	0.88 ± 0.11	0.91 ± 0.11	0.90 ± 0.13	0.87 ± 0.15	0.87 ± 0.14	0.89 ± 0.13	0.90 ± 0.08

Appendix B

Miscellaneous Information

Publications

Oreiller, V., Andrearczyk, V., Jreige, M., Boughdad, S., Elhalawani, H., Castelli, J., Vallières, M., Zhu, S., Xie, J., Peng, Y., **Iantsen, A.**, Hatt, M., Yuan, Y., Ma, J., Yang, X., Rao, C., Pai, S., Ghimire, K., Feng, X., Naser, M. A., Fuller, C. D., Yousefirizi, F., Rahmim, A., Chen, H., Wang, L., Prior, J. O., Depeursinge, A., "Head and Neck Tumor Segmentation in PET/CT: The HECKTOR Challenge". In: *Medical Image Analysis*, 2021.

Sepehri, S., Tankyevych, O., **Iantsen, A.**., Visvikis, D., Hatt, M., "Accurate tumor delineation vs. rough volume of interest analysis for 18F-FDG PET/CT radiomic-based prognostic modeling in Non-Small Cell Lung cancer". In: *Frontiers in Oncology*, 2021.

Iantsen, A., Ferreira, M., Lucia, F., Jaouen, V., Reinhold, C., Bonaffini, P., Alfieri, J., Rovira, R., Masson, I., Robin, P., Mervoyer, A., Rousseau, C., Kridelka, F., Decuypere, M., Lovinfosse, P., Pradier, O., Hustinx, R., Schick, U., Visvikis, D., Hatt, M., "Convolutional neural networks for PET functional volume fully automatic segmentation: development and validation in a multi-center setting". In: *European Journal of Nuclear Medicine and Molecular Imaging* 48.11, 2021.

Iantsen, A., Jaouen, V., Visvikis, D., Hatt, M., "Squeeze-and-Excitation Normalization for Brain Tumor Segmentation". In: *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries.* Ed. by A. Crimi and S. Bakas. Cham: Springer International Publishing, 2021.

Iantsen, A., Visvikis, D., Hatt, M., "Squeeze-and-Excitation Normalization for Automated Delineation of Head and Neck Primary Tumors in Combined PET and CT Images". In: *Head and Neck Tumor Segmentation.* Ed. by V. Andrearczyk, V. Oreiller, and A. Depeursinge. Cham: Springer International Publishing, 2021.

Presentations at Conferences and Workshops

"Fully Automated Detection and Segmentation of Hypermetabolic Lesions in Pretherapeutic [18F]FDG PET / CT Images of Lymphoma and Sarcoidosis Patients", *EANM'21*, Virtual, Oct. 2021.

"Squeeze-and-Excitation Normalization for Automated Delineation of Head and Neck Primary Tumors in Combined PET and CT Images", *MICCAI 2020*, Virtual, Oct. 2020.

"Automated Cervical Primary Tumor Functional Volume Segmentation in PET Images", *EANM'19*, Barcelona, Spain, Oct. 2019.

"Automated cervical primary tumor functional volume segmentation in PET images", *Imaging of diagnostic and therapeutic biomarkers in Oncology, workshop in Le Bono*, France, Sep. 2019.

Funding Details

The author received a PhD grant in the context of the Marie Skłodowska-Curie Innovative Training Network PREDICT [grant agreement no. 766276].

Bibliography

[1] H. Aerts et al. "Decoding tumour phenotype by noninvasive imaging using a quantitative radiomics approach". Undefined/Unknown. In: *Nature Communications* 5 (2014). ISSN: 2041-1723. DOI: 10.1038/ncomms5006.

[2] H. Akaike. "A new look at the statistical model identification". In: *IEEE Transactions on Automatic Control* 19.6 (1974), pp. 716–723. DOI: 10.1109/TAC.1974.1100705.

[3] D. M. Allen. "The Relationship Between Variable Selection and Data Agumentation and a Method for Prediction". In: *Technometrics* 16 (1974), pp. 125–127.

[4] M. Z. Alom et al. "Recurrent residual U-Net for medical image segmentation". In: *Journal of Medical Imaging* 6.1 (2019), pp. 1–16. DOI: 10.1117/1.JMI.6.1.014006. URL: https://doi.org/10.1117/1.JMI.6.1.014006.

[5] V. Andrearczyk et al. "Overview of the HECKTOR Challenge at MICCAI 2020: Automatic Head and Neck Tumor Segmentation in PET/CT". In: *Head and Neck Tumor Segmentation*. Ed. by V. Andrearczyk, V. Oreiller, and A. Depeursinge. Springer International Publishing, 2021, pp. 1–21.

[6] V. Andrearczyk, V. Oreiller, and A. Depeursinge, eds. *Head and Neck Tumor Segmentation - First Challenge, HECKTOR 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4, 2020, Proceedings*. Vol. 12603. Lecture Notes in Computer Science. Springer, 2021. ISBN: 978-3-030-67194-5. DOI: 10.1007/978-3-030-67194-5. URL: https://doi.org/10.1007/978-3-030-67194-5.

[7] M. Arbyn et al. "Estimates of Incidence and Mortality of Cervical Cancer in 2018: A Worldwide Analysis". In: *Lancet. Glob. Health* 8(2) (2020), pp. 191–203.

[8] M. Aristophanous et al. "A Gaussian Mixture Model for Definition of Lung Tumor Volumes in Positron Emission Tomography". In: *Med. Phys.* 34(11) (2007), pp. 4223–4235.

[9] J. L. Ba, J. R. Kiros, and G. E. Hinton. *Layer Normalization*. 2016. arXiv: 1607.06450 [stat.ML].

[10] V. Badrinarayanan, A. Kendall, and R. Cipolla. *SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation*. 2016. arXiv: 1511.00561 [cs.CV].

[11] D. Bahdanau, K. Cho, and Y. Bengio. "Neural Machine Translation by Jointly Learning to Align and Translate". In: *arXiv preprint arXiv:1409.0473* (2014).

[12] S. Bakas et al. "Identifying the Best Machine Learning Algorithms for Brain Tumor Segmentation, Progression Assessment, and Overall Survival Prediction in the BRATS Challenge". In: *arXiv preprint arXiv:1811.02629* (2018).

[13] S. Bakas et al. "Advancing The Cancer Genome Atlas glioma MRI collections with expert segmentation labels and radiomic features". In: *Scientific data* 4 (Sept. 2017), p. 170117. DOI: 10.1038/sdata.2017.117. URL: https://doi.org/10.1038/sdata.2017.117.

[14] S. Bakas et al. *Identifying the Best Machine Learning Algorithms for Brain Tumor Segmentation, Progression Assessment, and Overall Survival Prediction in the BRATS Challenge*. 2019. arXiv: 1811.02629 [cs.CV].

[15] S. F. Barrington et al. "Automated Segmentation of Baseline Metabolic Total Tumor Burden in Diffuse Large B-Cell Lymphoma: Which Method Is Most Successful? A Study on Behalf of the PETRA Consortium". In: *The Journal of Nuclear Medicine* 62 (2020), pp. 332–337.

[16] A. Barron. "Statistical properties of artificial neural networks". In: *Proceedings of the 28th IEEE Conference on Decision and Control,* 1989, 280–285 vol.1. DOI: 10.1109/CDC.1989.70117.

[17] S. Basu et al. "Implications of Standardized Uptake Value Measurements of the Primary Lesions in Proven Cases of Breast Carcinoma with Different Degree of Disease Burden at Diagnosis: Does 2-Deoxy-2-[F-18]fluoro-d-glucose-Positron Emission Tomography Predict Tumor Biology?" In: *Molecular Imaging and Biology* 10 (2007), pp. 62–66.

[18] Y. Bengio and Y. Grandvalet. "No Unbiased Estimator of the Variance of K-Fold Cross-Validation." In: *J. Mach. Learn. Res.* 5 (2004), pp. 1089–1105. URL: http://dblp.uni-trier.de/db/journals/jmlr/jmlr5.html#BengioG04.

[19] Y. Bengio and Y. Grandvalet. "No Unbiased Estimator of the Variance of K-Fold Cross-Validation." In: *J. Mach. Learn. Res.* 5 (2004), pp. 1089–1105.

[20] C. M. Bishop. "Training with Noise is Equivalent to Tikhonov Regularization". In: *Neural Computation* 7.1 (1995), pp. 108–116. DOI: 10.1162/neco.1995.7.1.108.

[21] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006. ISBN: 0387310738.

[22] P. Blanc-Durand et al. "Automatic Lesion Detection and Segmentation of 18F-FET PET in Gliomas: A Full 3D U-Net Convolutional Neural Network Study". In: *PLoS One* (2018).

[23] D. Bouget et al. *Meningioma segmentation in T1-weighted MRI leveraging global context and attention mechanisms*. 2021. arXiv: 2101.07715 [eess.IV].

[24] L. Breiman. "Bagging Predictors". In: *Machine Learning* 24.2 (1996), pp. 123–140.

[25] L. Breiman. "Random Forests". In: *Machine Learning* 45.1 (2001), pp. 5–32. ISSN: 0885-6125. DOI: 10.1023/A:1010933404324.

[26] L. Breiman. "Statistical modeling: the two cultures". In: *Statist. Sci.* 16.3 (2001), pp. 199–231. DOI: 10.1214/ss/1009213726.

[27] H. Brincker. "The sarcoidosis-lymphoma syndrome". In: *Br J Cancer* 54.3 (Sept. 1986), pp. 467–473.

[28] H. Brincker. "Interpretation of Granulomatous Lesions in Malignancy". In: *Acta Oncologica* 31.1 (1992), pp. 85–89. DOI: 10.3109/02841869209088273. eprint: https://doi.org/10.3109/02841869209088273. URL: https://doi.org/10.3109/02841869209088273.

[29] R. Caruana, S. Lawrence, and L. Giles. "Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping". In: *Advances in Neural Information Processing Systems 13 - Proceedings of the 2000 Conference, NIPS 2000*. 14th Annual Neural Information Processing Systems Conference, NIPS 2000 ; Conference date: 27-11-2000 Through 02-12-2000. Neural information processing systems foundation, 2001.

[30] L. Ceriani et al. “Utility of baseline 18FDG-PET/CT functional parameters in defining prognosis of primary mediastinal (thymic) large B-cell lymphoma”. In: *Blood* 126.8 (Aug. 2015), pp. 950–956. ISSN: 0006-4971. DOI: 10.1182/blood-2014-12-616474. eprint: https://ashpublications.org/blood/article-pdf/126/8/950/1391337/950.pdf. URL: https://doi.org/10.1182/blood-2014-12-616474.

[31] L. Ceriani et al. “Metabolic heterogeneity on baseline 18FDG-PET/CT scan is a predictor of outcome in primary mediastinal B-cell lymphoma”. In: *Blood* 132.2 (July 2018), pp. 179–186. ISSN: 0006-4971. DOI: 10.1182/blood-2018-01-826958. eprint: https://ashpublications.org/blood/article-pdf/132/2/179/1407217/blood826958.pdf. URL: https://doi.org/10.1182/blood-2018-01-826958.

[32] L. Chen et al. “Automatic PET Cervical Tumor Segmentation by Combining Deep Learning and Anatomic Prior”. In: *Phys. Med. Biol* (2019).

[33] Y. Chen et al. “Gradient descent with random initialization: fast global convergence for nonconvex phase retrieval”. In: *Mathematical Programming* 176.1-2 (Feb. 2019), pp. 5–37. ISSN: 1436-4646. DOI: 10.1007/s10107-019-01363-6. URL: http://dx.doi.org/10.1007/s10107-019-01363-6.

[34] Ö. Çiçek et al. *3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation*. 2016. arXiv: 1606.06650 [cs.CV].

[35] D. Ciresan et al. “Deep Neural Networks Segment Neuronal Membranes in Electron Microscopy Images”. In: *Advances in Neural Information Processing Systems*. Ed. by F. Pereira et al. Vol. 25. Curran Associates, Inc., 2012. URL: https://proceedings.neurips.cc/paper/2012/file/459a4ddcb586f24efd9395aa7662bc7c-Paper.pdf.

[36] D. Cireşan, U. Meier, and J. Schmidhuber. *Multi-column Deep Neural Networks for Image Classification*. 2012. arXiv: 1202.2745 [cs.CV].

[37] D. Cireşan et al. “A committee of neural networks for traffic sign classification”. In: *Neural Networks (IJCNN), The 2011 International Joint Conference on*. IEEE. San Jose, CA, 2011, pp. 1918–1921.

[38] D. C. Cireşan et al. “Flexible, High Performance Convolutional Neural Networks for Image Classification”. In: *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Two*. IJCAI’11. Barcelona, Catalonia, Spain: AAAI Press, 2011, pp. 1237–1242.

[39] D. C. Cireşan et al. “Mitosis Detection in Breast Cancer Histology Images with Deep Neural Networks”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2013*. Ed. by K. Mori et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 411–418. ISBN: 978-3-642-40763-5.

[40] N. Cohen, O. Sharir, and A. Shashua. *On the Expressive Power of Deep Learning: A Tensor Analysis*. 2016. arXiv: 1509.05009 [cs.NE].

[41] N. Coudray et al. “Classification and Mutation Prediction from Non-small Cell Lung Cancer Histopathology Images Using Deep Learning”. In: *Nat. Med.* 24(10) (2018), pp. 1559–1567.

[42] G. Creff et al. “Evaluation of the Prognostic Value of FDG PET/CT Parameters for Patients With Surgically Treated Head and Neck Cancer: A Systematic Review”. In: *JAMA Otolaryngol Head Neck Surg* 146.5 (May 2020), pp. 471–479.

[43] C. G. Cronin et al. “Clinical Utility of PET/CT in Lymphoma”. In: *American Journal of Roentgenology* 194.1 (2010). PMID: 20028897, W91–W103. DOI: 10.2214/AJR.09.2637. eprint: https://doi.org/10.2214/AJR.09.2637. URL: https://doi.org/10.2214/AJR.09.2637.

[44] G. Cybenko. “Approximation by superpositions of a sigmoidal function”. In: *Mathematics of Control, Signals, and Systems (MCSS)* 2 (1989), pp. 303–314. DOI: 10.1007/BF02551274.

[45] J. Deng et al. “ImageNet: a Large-Scale Hierarchical Image Database”. In: June 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.

[46] A. Depeursinge et al. “Predicting adenocarcinoma recurrence using computational texture models of nodule components in lung CT”. In: *Medical Physics* 42.4 (2015), pp. 2054–2063. DOI: https://doi.org/10.1118/1.4916088. eprint: https://aapm.onlinelibrary.wiley.com/doi/pdf/10.1118/1.4916088. URL: https://aapm.onlinelibrary.wiley.com/doi/abs/10.1118/1.4916088.

[47] M. Desseroit et al. “Development of a Nomogram Combining Clinical Staging with (18)F-FDG PET/CT Image Features in Non-small-cell Lung Cancer Stage I-III”. In: *Eur. J. Nucl. Med. Mol. Imaging.* 43(8) (2016), pp. 1477–1485.

[48] L. Devroye. “Automatic Pattern Recognition: A Study of the Probability of Error”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 10.4 (July 1988), pp. 530–543.

[49] T. G. Dietterich. "Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms". In: *Neural Computation* 10.7 (Oct. 1998), pp. 1895–1923. ISSN: 0899-7667. DOI: 10.1162/089976698300017197.

[50] T. G. Dietterich. "Ensemble Methods in Machine Learning". In: *Lecture Notes in Computer Science* 1857 (2000), pp. 1–15.

[51] J. Dolz et al. "HyperDense-Net: A Hyper-Densely Connected CNN for Multi-modal Image Segmentation". In: *IEEE Transactions on Medical Imaging* (2018).

[52] Q. Dou et al. *3D Deeply Supervised Network for Automatic Liver Segmentation from CT Volumes*. 2016. arXiv: 1607.00582 [cs.CV].

[53] J. Dubreuil et al. "Usual and unusual pitfalls of 18F-FDG-PET/CT in lymphoma after treatment: a pictorial review". In: *Nuclear medicine communications* 38 (May 2017), pp. 563–576. DOI: 10.1097/MNM.0000000000000697.

[54] J. F. Eary et al. "Spatial Heterogeneity in Sarcoma 18F-FDG Uptake as a Predictor of Patient Outcome". In: *Journal of Nuclear Medicine* 49.12 (2008), pp. 1973–1979. ISSN: 0161-5505. DOI: 10.2967/jnumed.108.053397. eprint: https://jnm.snmjournals.org/content/49/12/1973.full.pdf. URL: https://jnm.snmjournals.org/content/49/12/1973.

[55] B. Efron. "Bootstrap Methods: Another Look at the Jackknife". In: *The Annals of Statistics* 7.1 (1979), pp. 1–26. DOI: 10.1214/aos/1176344552.

[56] B. Efron. "Estimating the Error Rate of a Prediction Rule: Improvement on Cross-Validation". In: *Journal of the American Statistical Association* 78.382 (1983), pp. 316–331.

[57] B. Efron and R. J. Tibshirani. *An introduction to the bootstrap*. Chapman & Hall/CRC monographs on statistics and applied probability. 1993.

[58] T. El Jammal et al. "Sarcoidosis and Cancer: A Complex Relationship". In: *Frontiers in Medicine* 7 (2020), p. 857. ISSN: 2296-858X. DOI: 10.3389/fmed.2020.594118. URL: https://www.frontiersin.org/article/10.3389/fmed.2020.594118.

[59] I. El Naqa et al. "Concurrent Multimodality Image Segmentation by Active Contours for Radiotherapy Treatment Planning". In: *Med. Phys.* 34(12) (2007), pp. 4738–4749.

[60] R. Eldan and O. Shamir. "The Power of Depth for Feedforward Neural Networks". In: *29th Annual Conference on Learning Theory*. Ed. by V. Feldman, A. Rakhlin, and O. Shamir. Vol. 49. Proceedings of Machine Learning Research. Columbia University, New York, New York, USA: PMLR, June 2016, pp. 907–940. URL: https://proceedings.mlr.press/v49/eldan16.html.

[61] M. Everingham et al. "The Pascal Visual Object Classes (VOC) Challenge". In: *International Journal of Computer Vision* 88 (2009), pp. 303–338.

[62] F. Fathinul, A. J. Nordin, and W. F. Lau. "18[F]FDG-PET/CT is a useful molecular marker in evaluating tumour aggressiveness: a revised understanding of an in-vivo FDG-PET imaging that alludes the alteration of cancer biology". In: *Cell Biochem Biophys* 66.1 (May 2013), pp. 37–43.

[63] L. Freudenberg et al. "FDG-PET/CT in re-staging of patients with lymphoma". In: *European journal of nuclear medicine and molecular imaging* 31 (Mar. 2004), pp. 325–9. DOI: 10.1007/s00259-003-1375-y.

[64] R. Gatta et al. "Integrating radiomics into holomics for personalised oncology: from algorithms to bedside". In: *Eur Radiol Exp* 4.1 (Feb. 2020), p. 11.

[65] S. G. Gilmour. "The Interpretation of Mallows's C_p-Statistic". In: *Journal of the Royal Statistical Society. Series D (The Statistician)* 45.1 (1996), pp. 49–56.

[66] X. Glorot and Y. Bengio. "Understanding the difficulty of training deep feedforward neural networks". In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. Ed. by Y. W. Teh and M. Titterington. Vol. 9. Proceedings of Machine Learning Research. Chia Laguna Resort, Sardinia, Italy: PMLR, May 2010, pp. 249–256. URL: https://proceedings.mlr.press/v9/glorot10a.html.

[67] K. Gong et al. "PET Image Denoising Using a Deep Neural Network Through Fine Tuning". In: *IEEE Transactions on Radiation and Plasma Medical Sciences* 3(2) (2019), pp. 153–161.

[68] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[69] W. Grootjans et al. "Performance of Automatic Image Segmentation Algorithms for Calculating Total Lesion Glycolysis for Early Response Monitoring in Non-small Cell Lung Cancer Patients During Concomitant Chemoradiotherapy". In: *Radiother. Oncol.* 119(3) (2016), pp. 473–479.

[70] Z. Guo et al. “Deep Learning-Based Image Segmentation on Multimodal Medical Imaging”. In: *IEEE Transactions on Radiation and Plasma Medical Sciences* 3(2) (2019), pp. 162–169.

[71] Z. Guo et al. “Gross Tumor Volume Segmentation for Head and Neck Cancer Radiotherapy Using Deep Dense Multi-Modality Network”. In: *Phys. Med. Biol.* 64(20) (2019).

[72] B. Hanin and M. Sellke. *Approximating Continuous Functions by ReLU Nets of Minimal Width*. 2018. arXiv: 1710.11278 [stat.ML].

[73] L. Hansen and P. Salamon. “Neural network ensembles”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12.10 (1990), pp. 993–1001. DOI: 10.1109/34.58871.

[74] H. Hanzouli-Ben Salah et al. “A Framework Based on Hidden Markov Trees for Multimodal PET/CT Image Co-segmentation”. In: *Med. Phys.* 44(11) (2017), pp. 5835–5848.

[75] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Science & Business Media, 2009.

[76] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning: data mining, inference and prediction*. Springer, 2009. URL: http://www-stat.stanford.edu/~tibs/ElemStatLearn/.

[77] M. Hatt et al. “A Fuzzy Locally Adaptive Bayesian Segmentation Approach for Volume Determination in PET”. In: *IEEE Transactions on Medical Imaging* 28 (2009), pp. 881–893.

[78] M. Hatt et al. “Accurate Automatic Delineation of Heterogeneous Functional Volumes in Positron Emission Tomography for Oncology Applications”. In: *Int. J. Radiat. Oncol. Biol. Phys.* 77(1) (2010), pp. 301–308.

[79] M. Hatt et al. “Impact of Tumor Size and Tracer Uptake Heterogeneity in (18)F-FDG PET and CT Non-small-cell Lung Cancer Tumor Delineation”. In: *J. Nucl. Med.* 52(11) (2011), pp. 1690–1697.

[80] M. Hatt et al. “PET Functional Volume Delineation: A Robustness and Repeatability Study”. In: *Eur. J. Nucl. Med. Mol. Imaging* 38(4) (2011), pp. 663–672.

[81] M. Hatt et al. “18F-FDG PET uptake characterization through texture analysis: investigating the complementary nature of heterogeneity and functional tumor volume in a multi-cancer site patient cohort”. In: *J Nucl Med* 56.1 (Jan. 2015), pp. 38–44.

[82] M. Hatt et al. “Classification and Evaluation Strategies of Auto-segmentation Approaches for PET: Report of AAPM task group No. 211”. In: *Med. Phys.* 44(6) (2017), pp. 1–42.

[83] M. Hatt et al. “The First MICCAI Challenge on PET Tumor Segmentation”. In: *Medical Image Analysis* 44 (2018), pp. 177–195.

[84] M. Hatt et al. “Tumour Functional Sphericity from PET Images: Prognostic Value in NSCLC and Impact of Delineation Method”. In: *Eur. J. Nucl. Med. Mol. Imaging.* 45(4) (2018), pp. 630–641.

[85] M. Hatt et al. “Radiomics: Data Are Also Images”. In: *J. Nucl. Med.* 60(Suppl 2) (2019), pp. 38–44.

[86] S. S. Haykin. *Neural networks and learning machines*. Third. Pearson Education, 2009.

[87] K. He et al. “Deep Residual Learning for Image Recognition”. In: *Computer Vision and Pattern Recognition (CVPR)*. 2016.

[88] K. He et al. “Identity Mappings in Deep Residual Networks”. In: *European conference on Computer Vision (ECCV)*. 2016.

[89] K. He et al. *Deep Residual Learning for Image Recognition*. 2015. arXiv: 1512.03385 [cs.CV].

[90] K. He et al. “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Dec. 2015.

[91] K. He et al. *Identity Mappings in Deep Residual Networks*. 2016. arXiv: 1603.05027 [cs.CV].

[92] K. He et al. *Mask R-CNN*. 2018. arXiv: 1703.06870 [cs.CV].

[93] L. He et al. “The connected-component labeling problem: A review of state-of-the-art algorithms”. In: *Pattern Recognition* 70 (2017), pp. 25–43. ISSN: 0031-3203. DOI: https://doi.org/10.1016/j.patcog.2017.04.018. URL: https://www.sciencedirect.com/science/article/pii/S0031320317301693.

[94] N. Heller et al. *The state of the art in kidney and kidney tumor segmentation in contrast-enhanced CT imaging: Results of the KiTS19 Challenge*. 2020. arXiv: 1912.01054 [eess.IV].

[95] J. Hernandez, J. A. Carrasco-Ochoa, and J. F. Martínez-Trinidad. "An Empirical Study of Oversampling and Undersampling for Instance Selection Methods on Imbalance Datasets". In: *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*. Ed. by J. Ruiz-Shulcloper and G. Sanniti di Baja. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 262–269.

[96] T. Ho. "The Random Subspace Method for Constructing Decision Forests". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 20 (1998), pp. 832–844.

[97] K. Hornik, M. Stinchcombe, and H. White. "Multilayer feedforward networks are universal approximators". In: *Neural Networks* 2.5 (1989), pp. 359–366.

[98] K. Hornik. "Approximation capabilities of multilayer feedforward networks". In: *Neural Networks* 4.2 (1991), pp. 251–257. DOI: 10.1016/0893-6080(91)90009-T.

[99] J. Hu et al. *Squeeze-and-Excitation Networks*. 2019. arXiv: 1709.01507 [cs.CV].

[100] B. Huang et al. "Fully Automated Delineation of Gross Tumor Volume for Head and Neck Cancer on PET-CT Using Deep Learning: A Dual-Center Study". In: *Contrast Media & Molecular Imaging* (2018).

[101] G. Huang et al. "Densely Connected Convolutional Networks". In: *Proc. of the IEEE conference on Computer Vision and Pattern Recognition*. 2017, pp. 4700–4708.

[102] G. Huang et al. "Snapshot Ensembles: Train 1, get M for free". In: *CoRR* abs/1704.00109 (2017). arXiv: 1704.00109. URL: http://arxiv.org/abs/1704.00109.

[103] G. Huang et al. *Densely Connected Convolutional Networks*. 2018. arXiv: 1608.06993 [cs.CV].

[104] L. Huang et al. "Deep PET/CT Fusion with Dempster-Shafer Theory for Lymphoma Segmentation". In: *Machine Learning in Medical Imaging*. Ed. by C. Lian et al. Cham: Springer International Publishing, 2021, pp. 30–39. ISBN: 978-3-030-87589-3.

[105] A. Iantsen, D. Visvikis, and M. Hatt. "Squeeze-and-Excitation Normalization for Automated Delineation of Head and Neck Primary Tumors in Combined PET and CT Images". In: *HECKTOR 2020*. Ed. by V. Andrearczyk, V. Oreiller, and A. Depeursinge. Springer Nature Switzerland AG 2021, 2021, pp. 1–7.

[106] A. Iantsen et al. "Squeeze-and-Excitation Normalization for Brain Tumor Segmentation". In: *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*. Ed. by A. Crimi and S. Bakas. Cham: Springer International Publishing, 2021, pp. 366–373.

[107] H.-J. Im et al. "Current Methods to Define Metabolic Tumor Volume in Positron Emission Tomography: Which One is Better?" eng. In: *Nuclear medicine and molecular imaging* 52.1 (Feb. 2018). 493[PII], pp. 5–15. ISSN: 1869-3474. DOI: 10.1007/s13139-017-0493-6. URL: https://doi.org/10.1007/s13139-017-0493-6.

[108] S. Ioffe and C. Szegedy. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift". In: *Proceedings of the 32nd International Conference on Machine Learning*. Ed. by F. Bach and D. Blei. Vol. 37. Proceedings of Machine Learning Research. Lille, France: PMLR, July 2015, pp. 448–456. URL: https://proceedings.mlr.press/v37/ioffe15.html.

[109] F. Isensee and K. H. Maier-Hein. "An Attempt at Beating the 3D U-Net". In: *arXiv preprint arXiv:1908.02182* (2019).

[110] F. Isensee et al. "No New-Net". In: *arXiv preprint arXiv:1809.10483* (2018).

[111] F. Isensee and K. H. Maier-Hein. *An attempt at beating the 3D U-Net*. 2019. arXiv: 1908.02182 [eess.IV].

[112] F. Isensee et al. *nnU-Net for Brain Tumor Segmentation*. 2020. arXiv: 2011.00848 [eess.IV].

[113] F. Isensee et al. "nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation". In: *Nature Methods* 18.2 (Feb. 2021), pp. 203–211. ISSN: 1548-7105. DOI: 10.1038/s41592-020-01008-z. URL: https://doi.org/10.1038/s41592-020-01008-z.

[114] F. Isensee et al. "nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation". In: *Nature methods* 18.2 (Feb. 2021), pp. 203–211. ISSN: 1548-7091. DOI: 10.1038/s41592-020-01008-z. URL: https://doi.org/10.1038/s41592-020-01008-z.

[115] D. Jha et al. *ResUNet++: An Advanced Architecture for Medical Image Segmentation*. 2019. arXiv: 1911.07067 [eess.IV].

[116] H. Jia et al. *H2NF-Net for Brain Tumor Segmentation using Multimodal MR Imaging: 2nd Place Solution to BraTS Challenge 2020 Segmentation Task*. 2020. arXiv: 2012.15318 [eess.IV].

[117] Z. Jiang et al. "Two-Stage Cascaded U-Net: 1st Place Solution to BraTS Challenge 2019 Segmentation Task". In: *BrainLes@MICCAI*. 2019.

[118] Q. Jin et al. "RA-UNet: A Hybrid Deep Attention-Aware Network to Extract Liver and Tumor in CT Scans". In: *Frontiers in Bioengineering and Biotechnology* 8 (2020), p. 1471. ISSN: 2296-4185. DOI: 10.3389/fbioe.2020.605132. URL: https://www.frontiersin.org/article/10.3389/fbioe.2020.605132.

[119] K. Kamnitsas et al. "Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation". In: *Medical Image Analysis* 36 (Feb. 2017), pp. 61–78. ISSN: 1361-8415. DOI: 10.1016/j.media.2016.10.004. URL: http://dx.doi.org/10.1016/j.media.2016.10.004.

[120] B. Kayalibay, G. Jensen, and P. van der Smagt. *CNN-based Segmentation of Medical Imaging Data*. 2017. arXiv: 1701.03056 [cs.CV].

[121] N. S. Keskar et al. *On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima*. 2017. arXiv: 1609.04836 [cs.LG].

[122] D. P. Kingma and J. Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: 1412.6980 [cs.LG].

[123] B. Kocak et al. "Influence of Segmentation Margin on Machine Learning-Based High-Dimensional Quantitative CT Texture Analysis: A Reproducibility Study on Renal Clear Cell Carcinomas". In: *Eur. Radiol.* 29(9) (2019), pp. 4765–4775.

[124] R. Kohavi. "A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection". In: Morgan Kaufmann, 1995, pp. 1137–1143.

[125] Z. Kong et al. "18F-FDG-PET-based radiomics features to distinguish primary central nervous system lymphoma from glioblastoma". In: *NeuroImage : Clinical* 23 (2019).

[126] S. Koyasu et al. “Prognostic Value of Pretreatment 18F-FDG PET/CT Parameters Including Visual Evaluation in Patients With Head and Neck Squamous Cell Carcinoma”. In: *American Journal of Roentgenology* 202.4 (2014). PMID: 24660716, pp. 851–858. DOI: 10.2214/AJR.13.11013. eprint: https://doi.org/10.2214/AJR.13.11013. URL: https://doi.org/10.2214/AJR.13.11013.

[127] A. Krizhevsky, I. Sutskever, and G. E. Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems*. Ed. by F. Pereira et al. Vol. 25. Curran Associates, Inc., 2012. URL: https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

[128] D. M. Kurtz et al. “Noninvasive monitoring of diffuse large B-cell lymphoma by immunoglobulin high-throughput sequencing”. In: *Blood* 125.24 (June 2015), pp. 3679–3687. ISSN: 0006-4971. DOI: 10.1182/blood-2015-03-635169. eprint: https://ashpublications.org/blood/article-pdf/125/24/3679/1387184/3679.pdf. URL: https://doi.org/10.1182/blood-2015-03-635169.

[129] Y. LeCun et al. “Backpropagation Applied to Handwritten Zip Code Recognition”. In: *Neural Comput.* 1.4 (Dec. 1989), pp. 541–551. ISSN: 0899-7667. DOI: 10.1162/neco.1989.1.4.541. URL: https://doi.org/10.1162/neco.1989.1.4.541.

[130] Y. LeCun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE*. 1998, pp. 2278–2324.

[131] C.-Y. Lee et al. “Deeply-Supervised Nets”. In: *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*. Ed. by G. Lebanon and S. V. N. Vishwanathan. Vol. 38. Proceedings of Machine Learning Research. San Diego, California, USA: PMLR, May 2015, pp. 562–570. URL: https://proceedings.mlr.press/v38/lee15a.html.

[132] H. Lee et al. “Deep-Neural-Network-Based Sinogram Synthesis for Sparse-View CT Image Reconstruction”. In: *IEEE Transactions on Radiation and Plasma Medical Sciences* 3(2) (2019), pp. 109–119.

[133] M. Leshno et al. “Multilayer feedforward networks with a nonpolynomial activation function can approximate any function”. In: *Neural Networks* 6.6

(1993), pp. 861–867. ISSN: 0893-6080. DOI: https://doi.org/10.1016/S0893-6080(05)80131-5.

[134] K. H. Leung et al. "A Physics-Guided Modular Deep-Learning Based Automated Framework for Tumor Segmentation in PET". In: *arXiv preprint arXiv:1409.0473* (2020).

[135] L. Li et al. "Variational PET/CT Tumor Co-Segmentation Integrated with PET Restoration". In: *IEEE Transactions on Radiation and Plasma Medical Sciences* 4(1) (2019), pp. 37–49.

[136] L. Li et al. "IterNet: Retinal Image Segmentation Utilizing Structural Redundancy in Vessel Networks". In: *The IEEE Winter Conference on Applications of Computer Vision*. 2020.

[137] H. Lin and S. Jegelka. *ResNet with one-neuron hidden layers is a Universal Approximator*. 2018. arXiv: 1806.10909 [cs.LG].

[138] T.-Y. Lin et al. "Microsoft COCO: Common Objects in Context". In: *CoRR* abs/1405.0312 (2014). arXiv: 1405.0312. URL: http://arxiv.org/abs/1405.0312.

[139] T.-Y. Lin et al. "Focal Loss for Dense Object Detection". In: *2017 IEEE International Conference on Computer Vision (ICCV)*. 2017, pp. 2999–3007. DOI: 10.1109/ICCV.2017.324.

[140] M. Lippi et al. "Texture analysis and multiple-instance learning for the classification of malignant lymphomas". In: *Computer Methods and Programs in Biomedicine* 185 (2020), p. 105153. ISSN: 0169-2607. DOI: https://doi.org/10.1016/j.cmpb.2019.105153. URL: https://www.sciencedirect.com/science/article/pii/S0169260719302056.

[141] G. Litjens et al. "A Survey on Deep Learning in Medical Image Analysis". In: *Medical Image Analysis* 42 (2017), pp. 60–68.

[142] S. Lo et al. "Artificial convolution neural network techniques and applications for lung nodule detection". In: *IEEE transactions on medical imaging* 14.4 (1995), pp. 711–718. ISSN: 0278-0062. DOI: 10.1109/42.476112. URL: https://doi.org/10.1109/42.476112.

[143] J. London et al. "Sarcoidosis occurring after lymphoma: report of 14 patients and review of the literature". In: *Medicine (Baltimore)* 93.21 (Nov. 2014), e121.

[144] J. Long, E. Shelhamer, and T. Darrell. "Fully convolutional networks for semantic segmentation". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 3431–3440. DOI: 10.1109/CVPR.2015.7298965.

[145] N. M. Long and C. S. Smith. "Causes and imaging features of false positives and false negatives on F-PET/CT in oncologic imaging". In: *Insights Imaging* 2.6 (Dec. 2011), pp. 679–698.

[146] I. Loshchilov and F. Hutter. *SGDR: Stochastic Gradient Descent with Warm Restarts*. 2017. arXiv: 1608.03983 [cs.LG].

[147] Z. Lu et al. *The Expressive Power of Neural Networks: A View from the Width*. 2017. arXiv: 1709.02540 [cs.LG].

[148] F. Lucia et al. "External Validation of a Combined PET and MRI Radiomics Model for Prediction of Recurrence in Cervical Cancer Patients Treated with Chemoradiotherapy". In: *Eur. J. Nucl. Med. Mol. Imaging.* 46(4) (2019), pp. 864–877.

[149] K.-H. Lue et al. "Intratumor Heterogeneity Assessed by 18F-FDG PET/CT Predicts Treatment Response and Survival Outcomes in Patients with Hodgkin Lymphoma". In: *Academic Radiology* 27 (Nov. 2019), e183–e192. DOI: 10.1016/j.acra.2019.10.015.

[150] W. Luo et al. *Understanding the Effective Receptive Field in Deep Convolutional Neural Networks*. 2017. arXiv: 1701.04128 [cs.CV].

[151] M. Majdoub et al. "Prognostic Value of Head and Neck Tumor Proliferative Sphericity from 3'-Deoxy-3'-[18F] Fluorothymidine Positron Emission Tomography". In: *IEEE Transactions on Radiation and Plasma Medical Sciences* 2 (2018), pp. 33–40.

[152] C. L. Mallows. "Some Comments on C_P". In: *Technometrics* 15.4 (1973), pp. 661–675.

[153] Q. Martín-Saladich et al. "Comparison of different automatic methods for the delineation of the total metabolic tumor volume in I–II stage Hodgkin Lymphoma". In: *Scientific Reports* 10.1 (July 2020), p. 12590. ISSN: 2045-2322. DOI: 10.1038/s41598-020-69577-9. URL: https://doi.org/10.1038/s41598-020-69577-9.

[154] S. Martins et al. "Atlas-Based Multiorgan Segmentation for Dynamic Abdominal PET". In: *Med. Phys.* 46(11) (2019), pp. 4940–4950.

[155] M. Meignan et al. “Metabolic tumour volumes measured at staging in lymphoma: methodological evaluation on phantom experiments and patients”. In: *Eur J Nucl Med Mol Imaging* 41.6 (June 2014), pp. 1113–1122.

[156] B. H. Menze et al. “The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS)”. In: *IEEE Transactions on Medical Imaging* 34.10 (2015), pp. 1993–2024. DOI: 10.1109/TMI.2014.2377694.

[157] S. A. Milgrom et al. “A PET Radiomics Model to Predict Refractory Mediastinal Hodgkin Lymphoma”. eng. In: *Scientific reports* 9.1 (Feb. 2019), pp. 1322–1322. ISSN: 2045-2322. DOI: 10.1038/s41598-018-37197-z. URL: https://doi.org/10.1038/s41598-018-37197-z.

[158] F. Milletari, N. Navab, and S.-A. Ahmadi. *V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation*. 2016. arXiv: 1606.04797 [cs.CV].

[159] F. Mosteller and J. W. Tukey. “Data analysis, including statistics”. In: *Handbook of Social Psychology, Vol. 2*. Ed. by G. Lindzey and E. Aronson. Addison-Wesley, 1968.

[160] K. P. Murphy. *Machine learning : a probabilistic perspective*. Cambridge, Mass. [u.a.]: MIT Press, 2013.

[161] A. Myronenko. “3D MRI Brain Tumor Segmentation Using Autoencoder Regularization”. In: *arXiv preprint arXiv:1810.11654* (2018).

[162] C. Nadeau and Y. Bengio. “Inference for the Generalization Error”. In: *Advances in Neural Information Processing Systems*. Ed. by S. Solla, T. Leen, and K. Müller. Vol. 12. MIT Press, 2000. URL: https://proceedings.neurips.cc/paper/1999/file/7d12b66d3df6af8d429c1a357d8b9e1a-Paper.pdf.

[163] V. Nair and G. E. Hinton. “Rectified Linear Units Improve Restricted Boltzmann Machines”. In: *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*. Ed. by J. Fürnkranz and T. Joachims. 2010, pp. 807–814.

[164] H. Noh et al. “Regularizing Deep Neural Networks by Noise: Its Interpretation and Optimization”. In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper/2017/file/217e342fc01668b10cb1188d40d3370e-Paper.pdf.

[165] O. Oktay et al. “Attention U-net: Learning Where to Look for the Pancreas”. In: *arXiv preprint arXiv:1804.03999* (2018).

[166] D. Opitz and R. Maclin. “Popular Ensemble Methods: An Empirical Study”. In: *Journal of Artificial Intelligence Research* 11 (Aug. 1999), pp. 169–198. ISSN: 1076-9757. DOI: 10.1613/jair.614. URL: http://dx.doi.org/10.1613/jair.614.

[167] V. Oreiller et al. “Head and Neck Tumor Segmentation in PET/CT: The HECKTOR Challenge”. In: *Medical Image Analysis* (2021).

[168] F. Orlhac et al. “Tumor texture analysis in 18F-FDG PET: relationships between texture parameters, histogram indices, standardized uptake values, metabolic volumes, and total lesion glycolysis”. In: *J Nucl Med* 55.3 (Mar. 2014), pp. 414–422.

[169] X. Ou et al. “Ability of 18F-FDG PET/CT Radiomic Features to Distinguish Breast Carcinoma from Breast Lymphoma”. In: *Contrast Media & Molecular Imaging* 2019 (2019).

[170] X. Ou et al. “Radiomics based on 18F-FDG PET/CT could differentiate breast carcinoma from breast lymphoma using machine-learning approach: A preliminary study”. In: *Cancer Medicine* 9.2 (2020), pp. 496–506. DOI: https://doi.org/10.1002/cam4.2711. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/cam4.2711. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/cam4.2711.

[171] I. Papanikolaou and O. Sharma. “The relationship between sarcoidosis and lymphoma”. In: *European Respiratory Journal* 36.5 (2010), pp. 1207–1219. ISSN: 0903-1936. DOI: 10.1183/09031936.00043010. eprint: https://erj.ersjournals.com/content/36/5/1207.full.pdf. URL: https://erj.ersjournals.com/content/36/5/1207.

[172] B. Parmanto, P. Munro, and H. Doyle. “Improving Committee Diagnosis with Resampling Techniques”. In: *Advances in Neural Information Processing Systems*. Ed. by D. Touretzky, M. C. Mozer, and M. Hasselmo. Vol. 8. MIT Press, 1996. URL: https://proceedings.neurips.cc/paper/1995/file/46072631582fc240dd2674a7d063b040-Paper.pdf.

[173] E. Pfaehler et al. “Repeatability of 18 F-FDG PET Radiomic Features: A Phantom Study to Explore Sensitivity to Image Reconstruction Settings, Noise, and Delineation Method”. In: *Med. Phys.* 46(2) (2019), pp. 665–678.

[174] A. Pinkus. “Approximation theory of the MLP model in neural networks”. In: *ACTA NUMERICA* 8 (1999), pp. 143–195.

[175] H. B. Prabhakar et al. “Imaging Features of Sarcoidosis on MDCT, FDG PET, and PET/CT”. In: *American Journal of Roentgenology* 190.3_supplement (2008). PMID: 18287458, S1–S6. DOI: 10.2214/AJR.07.7001. eprint: https://doi.org/10.2214/AJR.07.7001. URL: https://doi.org/10.2214/AJR.07.7001.

[176] N. Qian. “On the momentum term in gradient descent learning algorithms”. In: *Neural Networks* 12.1 (1999), pp. 145–151. ISSN: 0893-6080. DOI: https://doi.org/10.1016/S0893-6080(98)00116-6. URL: https://www.sciencedirect.com/science/article/pii/S0893608098001166.

[177] C. Qin et al. “Convolutional Recurrent Neural Networks for Dynamic MR Image Reconstruction”. In: *IEEE Trans. Med. Imaging.* 38(1) (2019), pp. 280–290.

[178] A. J. Ramon et al. “Improving Diagnostic Accuracy in Low-Dose SPECT Myocardial Perfusion Imaging with Convolutional Denoising Networks”. In: *IEEE Trans. Med. Imaging.* 3(2) (2020).

[179] J. Redmon and A. Farhadi. *YOLOv3: An Incremental Improvement*. 2018. arXiv: 1804.02767 [cs.CV].

[180] S. Ren et al. “Atlas-Based Multiorgan Segmentation for Dynamic Abdominal PET”. In: *IEEE Transactions on Radiation and Plasma Medical Sciences* 4(1) (2019), pp. 50–62.

[181] S. Ren et al. *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks*. 2016. arXiv: 1506.01497 [cs.CV].

[182] S. Reuzé et al. “Prediction of Cervical Cancer Recurrence Using Textural Features Extracted from 18F-FDG PET Images Acquired with Different Scanners”. In: *Oncotarget.* 8(26) (2017), pp. 43169–43179.

[183] O. Ronneberger, P. Fischer, and T. Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Ed. by N. Navab et al. Cham: Springer International Publishing, 2015, pp. 234–241.

[184] A. G. Roy, N. Navab, and C. Wachinger. “Concurrent Spatial and Channel ‘squeeze & excitation’ in Fully Convolutional Networks”. In: *International conference on Medical Image Computing and Computer-Assisted Intervention*. 2018, pp. 421–429.

[185] S. Ruder. *An overview of gradient descent optimization algorithms*. 2017. arXiv: 1609.04747 [cs.LG].

[186] D. Rumelhart, G. Hinton, and R. Williams. "Learning internal representations by error propagation". In: *Parallel distributed processing*. Ed. by D. Rumelhart and J. McClelland. Vol. 1. Cambridge, MA: MIT Press, 1986, pp. 318–362.

[187] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. "Learning Representations by Back-propagating Errors". In: *Nature* 323.6088 (1986), pp. 533–536. DOI: 10.1038/323533a0. URL: http://www.nature.com/articles/323533a0.

[188] O. Russakovsky et al. "ImageNet Large Scale Visual Recognition Challenge". In: (2014). URL: http://arxiv.org/abs/1409.0575.

[189] S. Sarji. "Physiological uptake in FDG PET simulating disease". In: *Biomedical imaging and intervention journal* 2 (Oct. 2006), e59. DOI: 10.2349/biij.2.4.e59.

[190] N. Scher et al. "F)-FDG PET/CT parameters to predict survival and recurrence in patients with locally advanced cervical cancer treated with chemoradiotherapy". In: *Cancer Radiother* 22.3 (May 2018), pp. 229–235.

[191] D. Scherer, A. Müller, and S. Behnke. "Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition". In: *Proceedings of the 20th International Conference on Artificial Neural Networks: Part III*. ICANN'10. Thessaloniki, Greece: Springer-Verlag, 2010, pp. 92–101.

[192] M. Schmidt. *Least Squares Optimization with L1-Norm Regularization*. 2005.

[193] H. Schöder et al. "Intensity of 18fluorodeoxyglucose uptake in positron emission tomography distinguishes between indolent and aggressive non-Hodgkin's lymphoma". In: *J Clin Oncol* 23.21 (July 2005), pp. 4643–4651.

[194] G. Schwarz. "Estimating the Dimension of a Model". In: *The Annals of Statistics* 6.2 (1978), pp. 461–464.

[195] M. H. Schwarzbach et al. "Prognostic significance of preoperative [18-F] fluorodeoxyglucose (FDG) positron emission tomography (PET) imaging in patients with resectable soft tissue sarcomas". In: *Ann Surg* 241.2 (Feb. 2005), pp. 286–294.

[196] A. Shammas, R. Lim, and M. Charron. "Pediatric FDG PET/CT: Physiologic Uptake, Normal Variants, and Benign Conditions". In: *RadioGraphics* 29.5 (2009). PMID: 19755606, pp. 1467–1486. DOI: 10.1148/rg.295085247. eprint: https://doi.org/10.1148/rg.295085247. URL: https://doi.org/10.1148/rg.295085247.

[197] L. G. Shapiro. "Connected Component Labeling and Adjacency Graph Construction". In: *Topological Algorithms for Digital Image Processing*. Ed. by T. Y. Kong and A. Rosenfeld. Vol. 19. Machine Intelligence and Pattern Recognition. North-Holland, 1996, pp. 1–30. DOI: https://doi.org/10.1016/S0923-0459(96)80011-5. URL: https://www.sciencedirect.com/science/article/pii/S0923045996800115.

[198] K. Simonyan and A. Zisserman. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 2015. arXiv: 1409.1556 [cs.CV].

[199] L. N. Smith. "Cyclical Learning Rates for Training Neural Networks". In: *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*. 2017, pp. 464–472. DOI: 10.1109/WACV.2017.58.

[200] L. N. Smith and N. Topin. "Super-convergence: very fast training of neural networks using large learning rates". In: *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*. Ed. by T. Pham. Vol. 11006. International Society for Optics and Photonics. SPIE, 2019, pp. 369–386. URL: https://doi.org/10.1117/12.2520589.

[201] M. Soret, S. L. Bacharach, and I. Buvat. "Partial-Volume Effect in PET Tumor Imaging". In: *Journal of Nuclear Medicine* 48.6 (2007), pp. 932–945. ISSN: 0161-5505. DOI: 10.2967/jnumed.106.035774. eprint: https://jnm.snmjournals.org/content/48/6/932.full.pdf. URL: https://jnm.snmjournals.org/content/48/6/932.

[202] A. Srivastava et al. *MSRF-Net: A Multi-Scale Residual Fusion Network for Biomedical Image Segmentation*. 2021. arXiv: 2105.07451 [eess.IV].

[203] N. Srivastava et al. "Dropout: A Simple Way to Prevent Neural Networks from Overfitting". In: *Journal of Machine Learning Research* 15.56 (2014), pp. 1929–1958. URL: http://jmlr.org/papers/v15/srivastava14a.html.

[204] P. Starkov et al. "The use of texture-based radiomics CT analysis to predict outcomes in early-stage non-small cell lung cancer treated with stereotactic ablative radiotherapy". In: *The British Journal of Radiology* 92.1094 (2019). PMID: 30457885, p. 20180228. DOI: 10.1259/bjr.20180228. eprint: https://doi.org/10.1259/bjr.20180228. URL: https://doi.org/10.1259/bjr.20180228.

[205] M. Stone. "Cross-validatory choice and assessment of statistical predictions". In: *Journal of the royal statistical society. Series B (Methodological)* (1974), pp. 111–147.

[206] H. Sung et al. "Global Cancer Statistics 2020: GLOBOCAN Estimates of Incidence and Mortality Worldwide for 36 Cancers in 185 Countries". In: *CA: A Cancer Journal for Clinicians* 71.3 (2021), pp. 209–249. DOI: https://doi.org/10.3322/caac.21660. eprint: https://acsjournals.onlinelibrary.wiley.com/doi/pdf/10.3322/caac.21660. URL: https://acsjournals.onlinelibrary.wiley.com/doi/abs/10.3322/caac.21660.

[207] S. Supiot et al. "Evaluation of Tumor Hypoxia Prior to Radiotherapy in Intermediate-Risk Prostate Cancer Using 18F-fluoromisonidazole PET/CT: A Pilot Study". In: *Oncotarget.* 9(11) (2018), pp. 10005–10015.

[208] C. Szegedy et al. *Going Deeper with Convolutions.* 2014. arXiv: 1409.4842 [cs.CV].

[209] M. Tan and Q. Le. "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks". In: *Proceedings of the 36th International Conference on Machine Learning.* Ed. by K. Chaudhuri and R. Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, June 2019, pp. 6105–6114.

[210] S. Thulasidasan et al. "On Mixup Training: Improved Calibration and Predictive Uncertainty for Deep Neural Networks". In: *Advances in Neural Information Processing Systems.* Ed. by H. Wallach et al. Vol. 32. Curran Associates, Inc., 2019. URL: https://proceedings.neurips.cc/paper/2019/file/36ad8b5f42db492827016448975cc22d-Paper.pdf.

[211] T. Tieleman and G. Hinton. *Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude.* COURSERA: Neural Networks for Machine Learning. 2012.

[212] J. Tompson et al. *Efficient Object Localization Using Convolutional Networks.* 2015. arXiv: 1411.4280 [cs.CV].

[213] A. Torralba and A. A. Efros. "Unbiased look at dataset bias". In: *CVPR 2011*. 2011, pp. 1521–1528. DOI: 10.1109/CVPR.2011.5995347.

[214] L. Tuggener, J. Schmidhuber, and T. Stadelmann. "Is it Enough to Optimize CNN Architectures on ImageNet?" In: *CoRR* abs/2103.09108 (2021). arXiv: 2103.09108. URL: https://arxiv.org/abs/2103.09108.

[215] D. Ulyanov, A. Vedaldi, and V. Lempitsky. *Instance Normalization: The Missing Ingredient for Fast Stylization*. 2017. arXiv: 1607.08022 [cs.CV].

[216] N. Upadhyay and A. D. Waldman. "Conventional MRI evaluation of gliomas". In: *The British Journal of Radiology* 84.special_issue_2 (2011). PMID: 22433821, S107–S111. DOI: 10.1259/bjr/65711810. eprint: https://doi.org/10.1259/bjr/65711810. URL: https://doi.org/10.1259/bjr/65711810.

[217] M. Vallières et al. "Radiomics strategies for risk assessment of tumour failure in head-and-neck cancer". In: *CoRR* abs/1703.08516 (2017). arXiv: 1703.08516. URL: http://arxiv.org/abs/1703.08516.

[218] V. N. Vapnik and A. Y. Chervonenkis. "On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities". In: *Theory of Probability and its Applications* XVI.2 (1971), pp. 264–280.

[219] V. N. Vapnik. *The nature of statistical learning theory*. Springer-Verlag New York, Inc., 1995. ISBN: 0-387-94559-8.

[220] R. Wahl et al. "From RECIST to PERCIST: Evolving considerations for PET response criteria in solid tumors". English (US). In: *Journal of Nuclear Medicine* 50.SUPPL. 1 (May 2009), 122S–150S. ISSN: 0161-5505. DOI: 10.2967/jnumed.108.057307.

[221] H. Wang et al. "Current status and quality of radiomics studies in lymphoma: a systematic review". In: *Eur Radiol* 30.11 (Nov. 2020), pp. 6228–6240.

[222] L. Wang, C. Xie, and N. Zeng. "RP-Net: A 3D Convolutional Neural Network for Brain Segmentation From Magnetic Resonance Imaging". In: *IEEE Access* 7 (2019), pp. 39670–39679.

[223] Y. Wang et al. *Modality-Pairing Learning for Brain Tumor Segmentation*. 2020. arXiv: 2010.09277 [eess.IV].

[224] D. Wei et al. "MitoEM Dataset: Large-scale 3D Mitochondria Instance Segmentation from EM Images". In: *Med Image Comput Comput Assist Interv* 12265 (Oct. 2020), pp. 66–76.

[225] M. L. Welch et al. “Vulnerabilities of radiomic signature development: The need for safeguards”. In: *Radiother Oncol* 130 (Jan. 2019), pp. 2–9.

[226] A. C. Wilson et al. “The Marginal Value of Adaptive Gradient Methods in Machine Learning”. In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper/2017/file/81b3833e2504647f9d794f7d7b9bf341-Paper.pdf.

[227] D. H. Wolpert. “Stacked Generalization”. In: *Neural Networks* 5 (1992), pp. 241–259.

[228] S. Woo et al. “CBAM: Convolutional Block Attention Module”. In: *Proc. of the European conference on Computer Vision (ECCV)*. 2018, pp. 3–19.

[229] K. Wu, E. Otoo, and A. Shoshani. “Optimizing connected component labeling algorithms”. In: *Medical Imaging 2005: Image Processing*. Ed. by J. M. Fitzpatrick and J. M. Reinhardt. Vol. 5747. International Society for Optics and Photonics. SPIE, 2005, pp. 1965–1976. DOI: 10.1117/12.596105. URL: https://doi.org/10.1117/12.596105.

[230] Y. Wu and K. He. “Group Normalization”. In: *arXiv preprint arXiv:1803.08494* (2018).

[231] Y. Wu and K. He. *Group Normalization*. 2018. arXiv: 1803.08494 [cs.CV].

[232] Y. Yao, L. Rosasco, and A. Caponnetto. “On Early Stopping in Gradient Descent Learning”. In: *Constructive Approximation* 26.2 (2007), pp. 289–315. DOI: 10.1007/s00365-006-0663-2. URL: https://doi.org/10.1007/s00365-006-0663-2.

[233] J. Yosinski et al. “How transferable are features in deep neural networks?” In: *Advances in Neural Information Processing Systems*. Ed. by Z. Ghahramani et al. Vol. 27. Curran Associates, Inc., 2014. URL: https://proceedings.neurips.cc/paper/2014/file/375c71349b295fbe2dcdca9206f20a06-Paper.pdf.

[234] Y. Yuan. *Automatic Brain Tumor Segmentation with Scale Attention Network*. 2020. arXiv: 2011.03188 [eess.IV].

[235] T. Zhang. “Statistical behavior and consistency of classification methods based on convex risk minimization”. In: *The Annals of Statistics* 32 (Mar. 2004), pp. 56–134.

[236] X. Zhang et al. “The Effects of Volume of Interest Delineation on MRI-based Radiomics Analysis: Evaluation with Two Disease Groups”. In: *Cancer Imaging.* 19(1) (2019).

[237] L. Zhao et al. “Automatic Nasopharyngeal Carcinoma Segmentation Using Fully Convolutional Networks with Auxiliary Paths on Dual-Modality PET-CT Images”. In: *J. Digit. Imaging.* 32(3) (2019), pp. 462–470.

[238] X. Zhao et al. “Tumor Co-segmentation in PET/CT Using Multi-Modality Fully Convolutional Neural Network”. In: *Phys. Med. Biol.* 64(1) (2018).

[239] Z. Zhong et al. “3D Fully Convolutional Networks for Co-segmentation of Tumors on PET-CT Images”. In: *Proc. IEEE Int. Symp. Biomed. Imaging.* Vol. 52(11). 2018, pp. 228–231.

[240] D.-X. Zhou. “Universality of deep convolutional neural networks”. In: *Applied and Computational Harmonic Analysis* 48.2 (2020), pp. 787–794. DOI: https://doi.org/10.1016/j.acha.2019.06.004.

[241] P. Zhou et al. *Towards Theoretically Understanding Why SGD Generalizes Better Than ADAM in Deep Learning.* 2020. arXiv: 2010.05627 [cs.LG].

[242] T. Zhou, S. Canu, and S. Ruan. “Automatic COVID-19 CT segmentation using U-Net integrated spatial and channel attention mechanism”. In: *International Journal of Imaging Systems and Technology* 31.1 (Nov. 2020), pp. 16–27. ISSN: 1098-1098. DOI: 10.1002/ima.22527. URL: http://dx.doi.org/10.1002/ima.22527.

[243] T. Zhou et al. “A Tri-Attention fusion guided multi-modal segmentation network”. In: *Pattern Recognition* (2021), p. 108417. ISSN: 0031-3203. DOI: https://doi.org/10.1016/j.patcog.2021.108417. URL: https://www.sciencedirect.com/science/article/pii/S0031320321005938.

[244] T. Zhou et al. “Latent Correlation Representation Learning for Brain Tumor Segmentation With Missing MRI Modalities”. In: *IEEE Transactions on Image Processing* 30 (2021), pp. 4263–4274. DOI: 10.1109/TIP.2021.3070752.

[245] Y. Zhou et al. “D-UNet: A Dimension-Fusion U Shape Network for Chronic Stroke Lesion Segmentation”. In: *arXiv preprint arXiv:1908.05104* (2019).

[246] Y. Zhou et al. “Prediction of Overall Survival and Progression-Free Survival by the 18F-FDG PET/CT Radiomic Features in Patients with Primary Gastric Diffuse Large B-Cell Lymphoma”. In: *Contrast media & molecular imaging* 2019 (2019), p. 5963607. ISSN: 1555-4309. DOI: 10.1155/2019/5963607. URL: https://europepmc.org/articles/PMC6875372.

[247] Z. Zhou et al. “UNet++: A Nested U-Net Architecture for Medical Image Segmentation”. In: *4th Workshop on Deep Learning in Medical Image Analysis*. 2018.

[248] H. Zou and T. Hastie. “Regularization and Variable Selection via the Elastic Net”. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 67.2 (2003), pp. 301–320.

DOCTORAT BIOLOGIE BRETAGNE SANTE LOIRE

Université de Bretagne Occidentale

Titre : Segmentation Automatique de Tumeurs en Imagerie Multimodale TEP / TDM / IRM

Mot clés : imagerie médicale, apprentissage profond, segmentation, tumeur, réseau neuronal

Résumé : Les systèmes de reconnaissance visuelle fondés sur des réseaux de neurones convolutionnels (RNC) ont le potentiel d'améliorer le processus de prise en charge des patients. La fiabilité et la précision de solutions utilisant des RNC sont le sujet de cette thèse, dans le contexte de la segmentation automatisée de tumeurs.

Quatre tâches dans différentes modalités d'images ont été considérées. (1) Un modèle pour la segmentation de tumeurs du col de l'utérus en imagerie TEP a été entraîné sur des références fiables dans un contexte multicentrique, obtenant des performances robustes en s'appuyant sur de simples techniques d'augmentation de données. Un module de norme «squeeze-and-excitation» a été introduit dans le contexte de la segmentation de tumeurs tête et cou sur des images TEP/TDM multicentriques, obtenant de bons résultats sans standardisation spécifique. (3) La même approche, avec des modifications mineures, a été employée dans le contexte de tumeurs cérébrales dans des séquences IRM. Les deux méthodes ont obtenu des résultats compétitifs dans des compétitions en 2020, sans nécessité d'ajustement spécifique. (4) Un modèle a été construit pour la détection et la segmentation de lésions de lymphomes et de sarcoïdoses en imagerie TEP/TDM, avec de bons résultats sur les pathologies considérés, confirmant ainsi le potentiel des RNCs dans un contexte de plusieurs pathologies différentes.

Les résultats présentés dans cette thèse représentent une avancée vers l'intégration de ces méthodes en routine clinique.

Title: Automated Tumor Segmentation in Multimodal PET / CT / MR Imaging

Keywords: medical imaging, deep learning, segmentation, tumor, neural network

Abstract: In healthcare, visual recognition systems built on convolutional neural networks (CNNs) have the potential to improve the patient management process. The reliability and accuracy of CNN-based solutions are central research subjects of this thesis, examined in the context of automated tumor segmentation.

Four practical tasks with different image modalities were considered. (1) A model for cervical cancer segmentation in PET was developed in a multi-center setting, learning on reliable ground-truth labels. This approach provided robust performance by relying on simple data augmentation techniques. (2) A computational unit, squeeze-and-excitation norm, was introduced for head and neck primary tumor segmentation in PET/CT scans from multiple centers. A model with these units made accurate delineations without any center-specific data standardization. (3) The same approach, with minor modifications, was employed to delineate glioma subregions in multi-sequence MRI. Both methods obtained competitive results in two public challenges in 2020, essentially without any task-specific adjustments. (4) A model for detection and segmentation of lymphoma and sarcoidosis lesions in PET/CT was built. It achieved good overall results for the considered disorders, supporting potential applications of CNNs in a multi-disease context.

The findings described in this thesis represent a few steps towards the integration of CNN-based methods into daily clinical practice.