



HAL
open science

Lymphoma segmentation from 3D PET-CT images using a deep evidential network

Ling Huang, Su Ruan, Pierre Decazes, Thierry Dencœux

► **To cite this version:**

Ling Huang, Su Ruan, Pierre Decazes, Thierry Dencœux. Lymphoma segmentation from 3D PET-CT images using a deep evidential network. *International Journal of Approximate Reasoning*, 2022, 149, pp.39-60. 10.1016/j.ijar.2022.06.007 . hal-03766589

HAL Id: hal-03766589

<https://hal.science/hal-03766589>

Submitted on 1 Sep 2022

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

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

Lymphoma segmentation from 3D PET-CT images using a deep evidential network

Ling Huang^{a,b}, Su Ruan^b, Pierre Decazes^c, Thierry Dencœux^{a,d}

^a*Heudiasyc, CNRS, Université de technologie de Compiègne, Compiègne, France*
^b*Quantif, LITIS, University of Rouen Normandy, Rouen, France*
^c*Department of Nuclear Medicine, Henri Becquerel Cancer Center, Rouen, France*
^d*Institut universitaire de France, Paris, France*

Abstract

An automatic evidential segmentation method based on Dempster-Shafer theory and deep learning is proposed to segment lymphomas from three-dimensional Positron Emission Tomography (PET) and Computed Tomography (CT) images. The architecture is composed of a deep feature-extraction module and an evidential layer. The feature extraction module uses an encoder-decoder framework to extract semantic feature vectors from 3D inputs. The evidential layer then uses prototypes in the feature space to compute a belief function at each voxel quantifying the uncertainty about the presence or absence of a lymphoma at this location. Two evidential layers are compared, based on different ways of using distances to prototypes for computing mass functions. The whole model is trained end-to-end by minimizing the Dice loss function. The proposed combination of deep feature extraction and evidential segmentation is shown to outperform the baseline UNet model as well as three other state-of-the-art models on a dataset of 173 patients.

Keywords: medical image analysis, Dempster-Shafer theory, evidence theory, belief functions, uncertainty quantification, deep learning

1. Introduction

Positron Emission Tomography - Computed Tomography (PET-CT) scanning is an effective imaging tool for lymphoma segmentation with application to clinical diagnosis and radiotherapy planning. The standardized uptake value (SUV), defined as the measured activity normalized for body weight and injected dose to remove variability in image intensity between patients, is widely used to locate and segment lymphomas thanks to its high sensitivity and specificity to the metabolic activity of tumors [1]. However, PET images have a low resolution and suffer from the partial volume effect blurring the contours of objects [2]. For that reason, CT images are usually used jointly with PET images because of their anatomical feature-representation capability and high resolution. Figure 1 shows 3D PET-CT views of a lymphoma patient. The lymphomas are marked in black as well as the brain and the bladder. As we can see from this figure, lymphomas vary in intensity distribution, shape, type, and number.

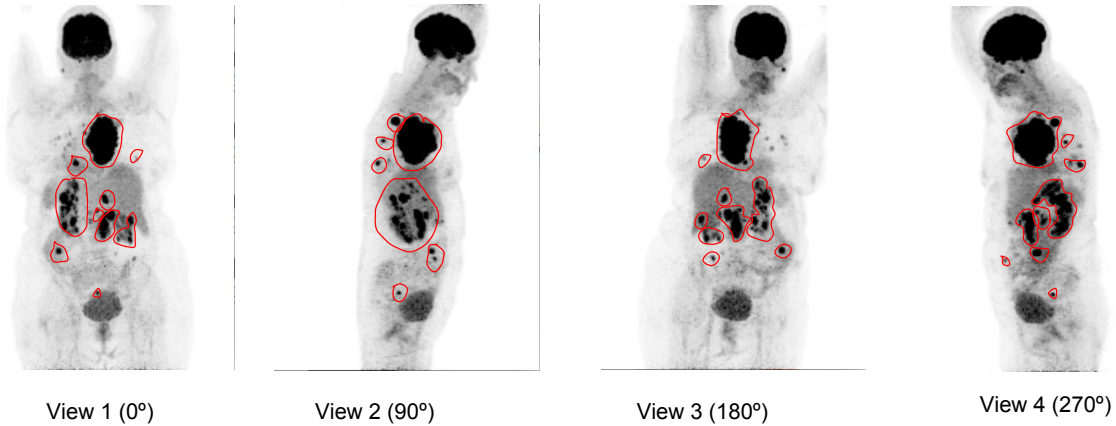


Figure 1: Example of a patient with lymphomas in 3D PET-CT views. The lymphomas are the dark areas circled in red.

14 *Approaches to lymphoma segmentation.* Techniques for lymphoma segmentation can be di-
 15 vided into three classes: SUV-based, region-growing-based and deep learning-based methods.
 16 For PET images, it is common to segment lymphomas with a set of fixed SUV thresholds.
 17 The so-called SUV-based methods [3][4] are fast but lack flexibility in boundary delineation
 18 and requires domain knowledge to locate the region of interest. Region-growing-based meth-
 19 ods [5][6] have been proposed to optimize boundary delineation by taking texture and shape
 20 information into account. By setting the specific growing function and stopping condition,
 21 the tumor region grows step by step until it reaches the stopping condition. However,
 22 those methods still need clinicians to locate the seeds for region growing [5] and they are
 23 time-consuming, especially when applied to 3D images. Lymphoma segmentation with deep
 24 learning has become a popular research topic thanks to its high feature representation ability
 25 [7][8].

26 *Deep-learning-based methods.* Long et al. [9] were the first authors to show that a fully con-
 27 volutional network (FCN) could be trained end-to-end for semantic segmentation, exceeding
 28 the state-of-the-art when the paper was published. UNet [10], a successful modification and
 29 extension of FCN, has become the most popular model for medical image segmentation
 30 in recent years. Driven by different tasks and datasets, several extended and optimized
 31 variants of UNet have been proposed for medical image segmentation, including VNet [11],
 32 SegResNet [12], and nnUNet [13]. VNet is a variant of UNet that introduces short residual
 33 connections at each stage. Compared with UNet, SegResNet contains an additional vari-
 34 ational autoencoder branch. Finally, nnUNet is more flexible than UNet in three aspects:
 35 (1) residual connection in convolution blocks, (2) anisotropic kernel sizes and strides in each
 36 layer, and (3) deep supervision heads. Deep learning has been applied to lymphoma segmen-
 37 tation, yielding promising results. In [7], Li et al. proposed a DenseX-Net-based lymphoma
 38 segmentation model with a two-flow architecture for 3D PET-CT images: a segmentation
 39 flow (DenseU-Net) for lymphoma segmentation and a reconstruction flow (encoder-decoder)
 40 for learning semantic representation of different lymphomas. In [8], Hu et al. introduced

41 a multi-source fusion model for lymphoma segmentation with PET images. First, three
42 2D and one 3D segmentation models were trained with three orthogonal views and one 3D
43 image, respectively. The four segmentation maps were then fused by a convolutional layer
44 to get a final result. In [14], Blanc-Durand et al. proposed a nnUNet-based lymphoma
45 segmentation network with additional validation of total metabolic tumor volume for 3D
46 PET-CT images. In [15], Huang et al. proposed to fuse the outputs of two UNets trained
47 on CT and PET data, using Dempster’s rule of combination [16], a combination operator of
48 Dempster-Shafer theory (DST) (see Section 2 below). However, the outputs of the UNets
49 were probabilities and this approach did not harness the full power of DST.

50 *Uncertainty.* In spite of the excellent performance of deep learning methods, the issue of
51 quantifying prediction uncertainty remains [17]. This uncertainty can be classified into
52 three types: distribution, model, and data uncertainty. Distribution uncertainty is caused
53 by training-test distribution mismatch (dataset shift) [18]. Model uncertainty arises from
54 limited training set size and model misspecification [19][20][21]. Finally, sources of data un-
55 certainty include class overlap, label noise, and homo or hetero-scedastic noise [22]. Because
56 of the limitations of medical imaging and labeling technology, as well as the need to use
57 a large nonlinear parametric segmentation model, PET-CT image segmentation results are
58 particularly tainted with uncertainty, which limits the reliability of the segmentation. Figure
59 2 shows examples of PET and CT image slices for one patient with lymphomas. As can
60 be seen, lymphomas in PET images usually correspond to the brightest pixels, but organs
61 such as the brain and bladder are also located in bright pixel areas, which may result in
62 segmentation errors. Moreover, lymphoma boundaries are blurred, which makes it hard to
63 delineate lymphomas precisely.

64 *Approaches to uncertainty modeling.* Early approaches to uncertainty quantification in ma-
65 chine learning were based on Bayesian theory [23][24]. The popularity of deep learning
66 models has revived research of model uncertainty estimation and has given rise to specific
67 methods such as variational dropout [25][26]. In this paper, we explore a different approach
68 based on DST [27][16] [28], a theoretical framework for reasoning with imperfect (uncertain,
69 imprecise, partial) information. DST was first introduced by Dempster [27] and Shafer [16]
70 and was further popularized and developed by Smets [29]. Applications in machine learning
71 were first introduced by Dencœux [30, 31, 32]. DST is based on the representation of ele-
72 mentary items of evidence by belief functions, and their combination by a specific operator
73 called Dempster’s rule of combination. In recent years, DST has generated considerable
74 interest and has had great success in various fields, including information fusion [33][34][35],
75 classification [36][37][38], clustering [28][39][40], and image segmentation [41][42][43].

76 In this paper¹, we propose a 3D PET-CT diffuse large B-cell lymphoma segmentation
77 model based on DST and deep learning, which not only focuses on lymphoma segmenta-
78 tion accuracy but also on uncertainty quantification using belief functions. The proposed

¹This paper is an extended version of the short paper presented at the 6th International Conference on Belief Functions (BELIEF 2021) [44].

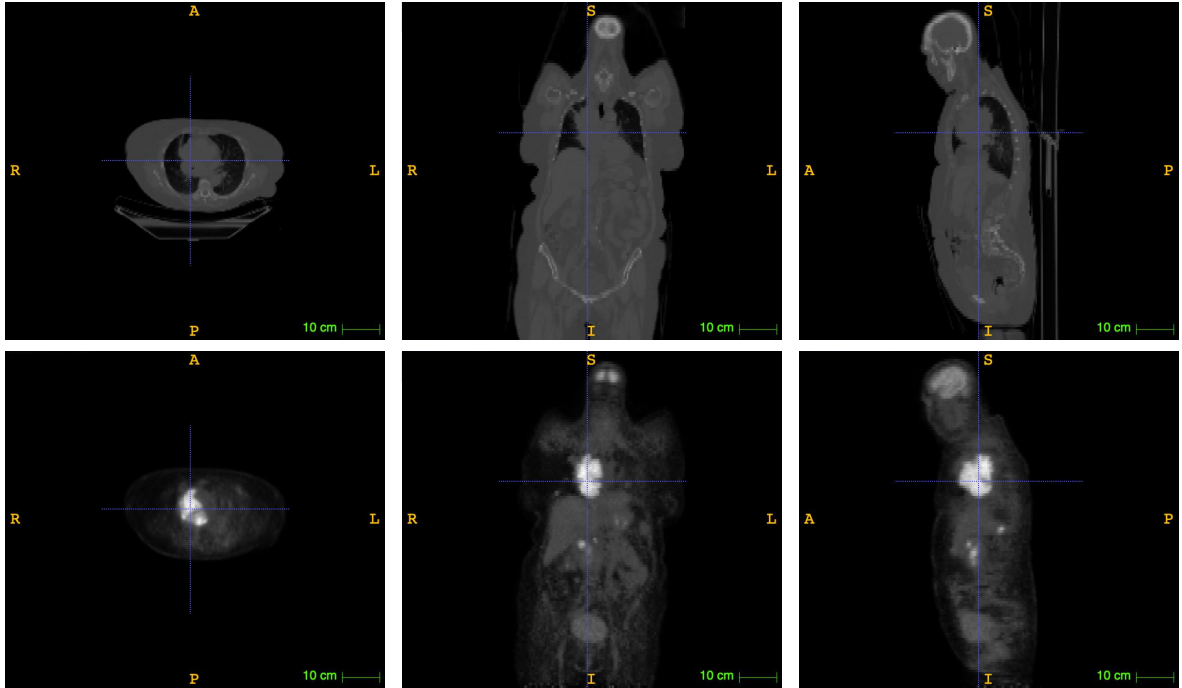


Figure 2: Example of a patient with lymphomas. The first and second rows show, respectively, PET and CT slices for one patient in axial, sagittal and coronal views.

79 segmentation model is composed of a UNet module for feature extraction and an eviden-
 80 tial segmentation module for uncertainty quantification and decision-making. End-to-end
 81 learning is performed by minimizing the Dice loss function.

82 The rest of the paper is organized as follows. The main concepts of DST are first recalled
 83 in Section 2, and two approaches for computing belief functions in classification tasks are
 84 described in Section 3. The proposed model is then introduced in Section 4, and experimental
 85 results are reported in Section 5. Finally, Section 6 concludes the paper.

86 2. Dempster-Shafer theory

87 In this section, we first recall some necessary notations and definitions regarding DST.
 88 Let $\Omega = \{\omega_1, \omega_2, \dots, \omega_K\}$ be a finite set of all possible answers some question, called the
 89 *frame of discernment*. Evidence about the question of interest can be represented by a *mass*
 90 *function* m , defined as a mapping from the power set 2^Ω to $[0, 1]$ such that

$$\sum_{A \subseteq \Omega} m(A) = 1 \quad (1)$$

91 and $m(\emptyset) = 0$, where \emptyset denotes the empty set. Subsets $A \subseteq \Omega$ such $m(A) > 0$ are called the
 92 *focal sets* of m . Each mass $m(A)$ represents a share of a unit mass of belief allocated to focal
 93 set A , and which cannot be allocated to any strict subset of A . The mass $m(\Omega)$ allocated
 94 to the whole frame can be seen as a degree of ignorance. Full ignorance is represented by

95 the *vacuous* mass function m_γ verifying $m_\gamma(\Omega) = 1$. A mass function is said to be *Bayesian*
 96 if its focal sets are singletons, and *logical* if it has only one focal set.

97 *Discounting.* Let m be a mass function on Ω and s a coefficient in $[0, 1]$. The *discounting*
 98 operation [16] with discount rate $1 - s$ transforms m into a weaker, less informative mass
 99 function defined as follows:

$${}^s m = s m + (1 - s) m_\gamma. \quad (2)$$

100 As shown in [45], coefficient s can be interpreted as a degree of belief that the source of
 101 information providing mass function m is reliable.

102 *Simple mass functions.* A mass function m is said to be *simple* if it can be obtained by
 103 discounting a logical mass function; it thus has the following form:

$$m(A) = s, \quad m(\Omega) = 1 - s, \quad (3)$$

104 for some $A \subset \Omega$ such that $A \neq \emptyset$ and some $s \in [0, 1]$, called the *degree of support* in A .
 105 The quantity $w = -\ln(1 - s)$ is called the *weight of evidence* associated to m [16, page 77].
 106 In the following, a simple mass function with focal set A and weight of evidence w will be
 107 denoted as A^w .

108 *Belief and plausibility.* Given a mass function m , *belief* and *plausibility* functions are defined,
 109 respectively, as follows:

$$Bel(A) = \sum_{B \subseteq A} m(B) \quad (4)$$

110 and

$$Pl(A) = \sum_{B \cap A \neq \emptyset} m(B) = 1 - Bel(A^c), \quad (5)$$

111 for all $A \subseteq \Omega$, where A^c denotes the complement of A . The quantity $Bel(A)$ can be
 112 interpreted as a degree of support for A , while $Pl(A)$ can be interpreted as a measure of
 113 lack of support for the complement of A .

114 *Dempster's rule.* Two mass functions m_1 and m_2 derived from two independent items of
 115 evidence can be combined by considering each pair of a focal set B of m_1 and a focal set C
 116 of m_2 , and assigning the product $m_1(B)m_2(C)$ to the intersection $B \cap C$. A normalization
 117 step is then necessary to ensure that the mass of the empty set is equal to zero. This
 118 operation, called *Dempster's rule of combination* [16] and denoted as \oplus , is formally defined
 119 by $(m_1 \oplus m_2)(\emptyset) = 0$ and

$$(m_1 \oplus m_2)(A) = \frac{1}{1 - \kappa} \sum_{B \cap C = A} m_1(B)m_2(C), \quad (6)$$

120 for all $A \subseteq \Omega, A \neq \emptyset$, where κ represents the *degree of conflict* between m_1 and m_2 equal to

$$\kappa = \sum_{B \cap C = \emptyset} m_1(B)m_2(C). \quad (7)$$

121 The combined mass $m_1 \oplus m_2$ is called the *orthogonal sum* of m_1 and m_2 . It can easily be
 122 checked that the orthogonal sum of two simple mass functions A^{w_1} and A^{w_2} with the same
 123 focal set A is the simple mass function $A^{w_1+w_2}$: Dempster’s rule thus adds up weights of
 124 evidence.

125 *Decision-making.* After aggregating all the available evidence in the form of a mass function,
 126 it is often necessary to make a final decision. Decision-making based on belief functions for
 127 classification tasks has been studied in [46], and, more recently, by Ma and Dencœux in
 128 [47]. The reader is referred to Ref. [48] for a recent review of decision methods based on
 129 belief functions. Here, we briefly introduce the approach used in this paper. Consider a
 130 classification task with K classes in the set $\Omega = \{\omega_1, \dots, \omega_K\}$. Assume that the utility
 131 of selecting the correct class is 1, and the utility of an error is 0. As shown in [46], the
 132 lower and upper expected utilities of selecting class ω_k are then, respectively, $Bel(\{\omega_k\})$ and
 133 $Pl(\{\omega_k\})$. A pessimistic decision-maker (DM) maximizing the lower expected utility will
 134 then select the class with the highest degree of belief, while an optimistic DM minimizing
 135 the upper expected utility will select the most plausible class. Alternatively, the Hurwicz
 136 criterion consists in maximizing a weighted sum of the lower and upper expected utility. In
 137 the decision context, we then select the class ω_k such that $(1 - \xi)Bel(\{\omega_k\}) + \xi Pl(\{\omega_k\})$
 138 is maximum, where ξ is an optimism index. Another approach, advocated by Smets in the
 139 Transferable Belief Model [45], is to base decisions on the pignistic probability distribution,
 140 defined as

$$p_m(\omega) = \sum_{\{A \subseteq \Omega: \omega \in A\}} \frac{m(A)}{|A|} \quad (8)$$

141 for all $\omega \in \Omega$.

142 3. Evidential classifiers

143 In this section, we review two methods for designing classifiers that output mass func-
 144 tions, referred to as *evidential classifiers*. The evidential neural network (ENN) classifier
 145 introduced in [31] is first recalled in Section 3.1. A new model based on the interpretation
 146 of a radial basis function (RBF) network as combining of simple mass functions by Demp-
 147 ster’s rule, inspired by [49], is then described in Section 3.2. The two models are compared
 148 experimentally in Section 3.3.

149 3.1. Evidential neural network

150 In [31], Dencœux proposed the ENN classifier, in which mass functions are computed
 151 based on distances to prototypes. The basic idea is to consider each prototype as a piece of
 152 evidence, which is discounted based on its distance to the input vector. The evidence from
 153 different prototypes is then pooled by Dempster’s rule (6). We provide a brief introduction
 154 to the ENN model in this section.

155 The ENN classifier is composed on an input layer of H neurons (where H is the dimension
 156 of input space), two hidden layers and an output layer (Figure 3). The first input layer is

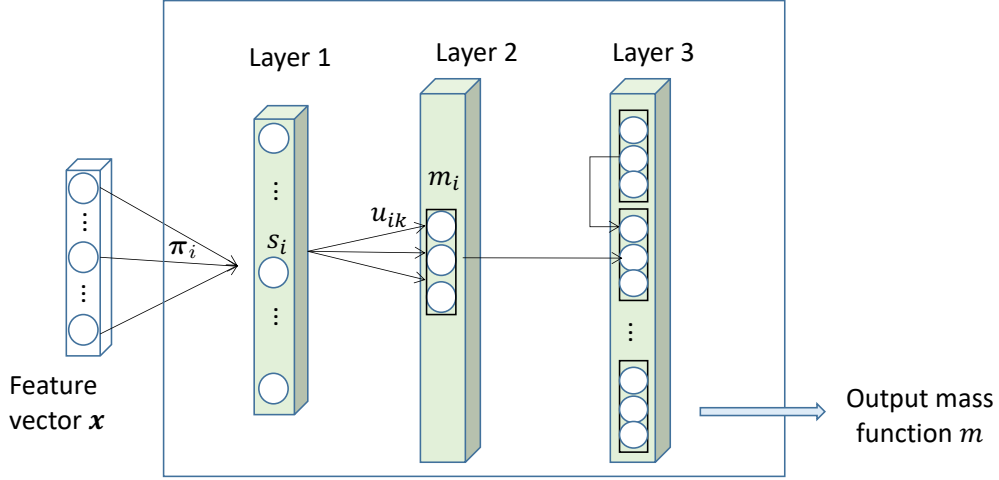


Figure 3: Evidential neural network.

157 composed of I units, whose weights vectors are prototypes π_1, \dots, π_I in input space. The
 158 activation of unit i in the prototype layer is

$$s_i = \alpha_i \exp(-\gamma_i d_i^2), \quad (9)$$

159 where $d_i = \|\mathbf{x} - \pi_i\|$ is the Euclidean distance between input vector \mathbf{x} and prototype π_i ,
 160 $\gamma_i > 0$ is a scale parameter, and $\alpha_i \in [0, 1]$ is an additional parameter.

The second hidden layer computes mass functions m_i representing the evidence of each prototype π_i , using the following equations:

$$m_i(\{\omega_k\}) = u_{ik} s_i, \quad k = 1, \dots, K \quad (10a)$$

$$m_i(\Omega) = 1 - s_i, \quad (10b)$$

161 where u_{ik} is the membership degree of prototype i to class ω_k , and $\sum_{k=1}^K u_{ik} = 1$. Mass
 162 function m_i can thus be seen as a discounted Bayesian mass function, with discount rate
 163 $1 - s_i$; its focal sets are singletons and Ω . The mass assigned to Ω increases with the distance
 164 between \mathbf{x} and π_i . Finally, the third layer combines the I mass functions m_1, \dots, m_I using
 165 Dempster's rule (6). The output mass function $m = \bigoplus_{i=1}^I m_i$ is a discounted Bayesian mass
 166 function that summarizes the evidence of the I prototypes. Because the focal sets of m are
 167 singletons and Ω , the class with the highest degree of belief also has the highest plausibility
 168 and pignistic probability: consequently, the decision rules recalled in Section 2 are equivalent
 169 in this case.

170 Let θ denote the vector of all network parameters, composed of the I prototypes π_i ,
 171 their parameters γ_i and α_i , and their membership degrees u_{ik} , $k = 1, \dots, K$. In [31], it
 172 was proposed to learn these parameters by minimizing the regularized sum-of-squares loss
 173 function

$$L_{SS}(\theta) = \sum_{n=1}^N \sum_{k=1}^K (p_{nk} - y_{nk})^2 + \lambda \sum_{i=1}^I \alpha_i, \quad (11)$$

174 where p_{nk} is the pignistic probability of class ω_k for instance n , N is the number of training
 175 instances, and $y_{nk} = 1$ if the true class of instance n is ω_k , and $y_{nk} = 0$ otherwise. The
 176 second term on the right-hand side of (11) is a regularization term, and λ is hyperparameter
 177 that can be tuned by cross-validation.

178 The idea of applying the above model to features extracted by a convolutional neural
 179 network (CNN) was first proposed by Tong et al. in [50]. In this approach, the ENN module
 180 becomes a “evidential layer”, which is plugged into the output of a CNN instead of the usual
 181 softmax layer. The feature extraction and evidential modules are trained simultaneously.
 182 A similar approach was applied in [43] to semantic segmentation. In the next section, we
 183 present an alternative approach based on a radial basis function (RBF) network and weights
 184 of evidence.

185 3.2. Radial basis function network

186 As shown in [49], the calculations performed in the softmax layer of a feedforward neural
 187 network can be interpreted in terms of combination of evidence by Dempster’s rule. The
 188 output class probabilities can be seen as normalized plausibilities according to an underlying
 189 belief function. Applying these ideas to a radial basis function (RBF) network, it is possible
 190 to derive an alternative evidential classifier with properties similar to those of the ENN
 191 model recalled in Section 3.1.

192 Consider an RBF network with I prototype (hidden) units. The activation of hidden
 193 unit i is

$$s_i = \exp(-\gamma_i d_i^2), \quad (12)$$

194 where, as before, $d_i = \|\mathbf{x} - \boldsymbol{\pi}_i\|$ is the Euclidean distance between input vector \mathbf{x} and
 195 prototype $\boldsymbol{\pi}_i$, and $\gamma_i > 0$ is a scale parameter. For the application considered in this paper,
 196 we only need to consider the case of binary classification with $K = 2$ and $\Omega = \{\omega_1, \omega_2\}$.
 197 (The case where $K > 2$ is also analyzed in [49]). Let v_i be the weight of the connection
 198 between hidden unit i and the output unit, and let $w_i = s_i v_i$ be the product of the output
 199 of unit i and weight v_i . The quantities w_i can be interpreted as weights of evidence for class
 200 ω_1 or ω_2 , depending on the sign of v_i :

- 201 • If $v_i \geq 0$, w_i a weight of evidence for class ω_1 ;
- 202 • If $v_i < 0$, $-w_i$ is a weight of evidence for class ω_2 .

203 To each prototype i can, thus, be associated the following simple mass function:

$$m_i = \{\omega_1\}^{w_i^+} \oplus \{\omega_2\}^{w_i^-},$$

204 where $w_i^+ = \max(0, w_i)$ and $w_i^- = -\min(0, w_i)$ denote, respectively, the positive and nega-
 205 tive parts of w_i . Combining the evidence of all prototypes in favor of ω_1 or ω_2 by Dempster’s
 206 rule, we get the mass function

$$m = \bigoplus_{i=1}^I m_i = \{\omega_1\}^{w^+} \oplus \{\omega_2\}^{w^-}, \quad (13)$$

207 with $w^+ = \sum_{i=1}^I w_i^+$ and $w^- = \sum_{i=1}^I w_i^-$. In [49], the normalized plausibility of ω_1 corre-
 208 sponding to mass function m was shown to have the following expression:

$$p(\omega_1) = \frac{Pl(\{\omega_1\})}{Pl(\{\omega_1\}) + Pl(\{\omega_2\})} = \frac{1}{1 + \exp(-\sum_{i=1}^I v_i s_i)}, \quad (14)$$

i.e., it is the output of a unit with a logistic activation function. When training an RBF network with a logistic output unit, we thus actually combine evidence from each of the prototypes, but the combined mass function remains latent. In [49], mass function m defined by (13) was shown to have the following expression:

$$m(\{\omega_1\}) = \frac{[1 - \exp(-w^+)] \exp(-w^-)}{1 - \kappa} \quad (15a)$$

$$m(\{\omega_2\}) = \frac{[1 - \exp(-w^-)] \exp(-w^+)}{1 - \kappa} \quad (15b)$$

$$m(\Omega) = \frac{\exp(-w^+ - w^-)}{1 - \kappa} = \frac{\exp(-\sum_{i=1}^I |w_i|)}{1 - \kappa}, \quad (15c)$$

209 where

$$\kappa = [1 - \exp(-w^+)][1 - \exp(-w^-)] \quad (15d)$$

210 is the degree of conflict between mass functions $\{\omega_1\}^{w^+}$ and $\{\omega_2\}^{w^-}$.

211 In the approach, we thus simply need to train a standard RBF network with I prototype
 212 layers and one output unit with a logistic activation function, by minimizing a loss function
 213 such as, e.g., the regularized cross-entropy loss

$$L_{CE}(\boldsymbol{\theta}) = -\sum_{n=1}^N (y_n \log p_n + (1 - y_n) \log(1 - p_n)) + \lambda \sum_{i=1}^I w_i^2, \quad (16)$$

214 where p_n is the normalized plausibility of class ω_1 computed from (14) for instance n , y_n is
 215 class label of instance n ($y_n = 1$ if the true class of instance n is ω_1 , and $y_n = 0$ otherwise),
 216 and λ is a hyperparameter. We note that increasing λ has the effect of decreasing the weights
 217 of evidence and, thus, obtaining less informative mass functions.

218 3.3. Comparison between the two models

219 To compare the RBF model described in Section 3.2 with the ENN model recalled in
 220 Section 3.1, we consider the two-class dataset shown in Figure 4. The two classes are ran-
 221 domly distributed around half circles with Gaussian noise and are separated by a nonlinear
 222 boundary. A learning set of size $N = 300$ and a test set of size 1000 were generated from
 223 the same distribution.

224 An ENN and a RBF network were initialized with $I = 6$ prototypes generated by the k -
 225 means algorithm and were trained on the learning data. Figures 5a and 5b show, respectively,
 226 the test error rate and the mean uncertainty (defined as the average mass assigned to the
 227 frame Ω), as functions of hyperparameter λ in (11) and (16), for 10 different runs of both

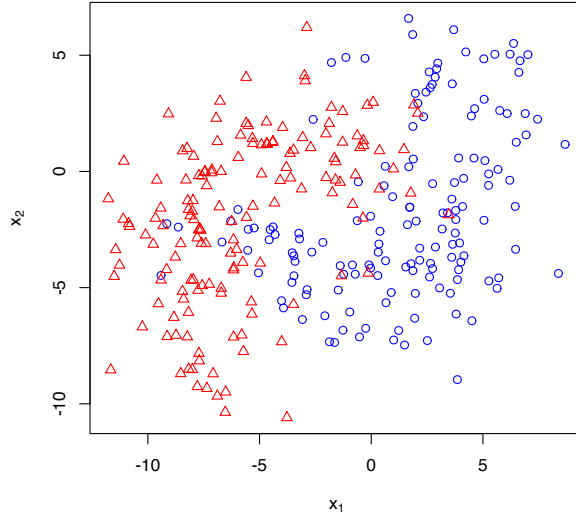


Figure 4: Simulated data.

228 algorithms with different initializations. As expected, uncertainty increases with λ for both
 229 models, but the ENN model appears to be less sensitive to λ as compared to the RBF model.
 230 Both models achieve similar minimum error rates for λ around 10^{-3} , and have similar mean
 231 uncertainties for $\lambda = 10^{-4}$.

232 As shown in [31], the robustness of the ENN model arises from the fact that, when the
 233 input \mathbf{x} is far from all prototypes, the output mass function m is close to the vacuous mass
 234 function. This property, in particular, makes the network capable of detecting observations
 235 generated from a distribution that is not represented in the learning set. From (15c), we can
 236 expect the RBF network model to have a similar property: if \mathbf{x} is far from all prototypes, all
 237 weights of evidence w_i will be small and the mass $m(\Omega)$ will be close to unity. To compare
 238 the mass functions computed by the two models, not only in regions of high density where
 239 training data are present, but also in regions of low density, we introduced a third class in
 240 the test set, as shown in Figure 6. Figure 7 shows scatter plots of masses on each of the focal
 241 sets computed for the two models trained with $\lambda = 10^{-3}$ and applied to an extended dataset
 242 composed of the learning data and the third class. We can see that the mass functions are
 243 quite similar. Contour plots shown in Figure 6 confirm this similarity.

244 4. Proposed model

245 The main idea of this work is to hybridize a deep medical image segmentation model
 246 with one of the evidential classifiers introduced in Section 3. Figure 8 shows the global
 247 lymphoma segmentation architecture, composed of an encoder-decoder feature extraction
 248 module (UNet), and an evidential layer based one of the two models described in Section 3.

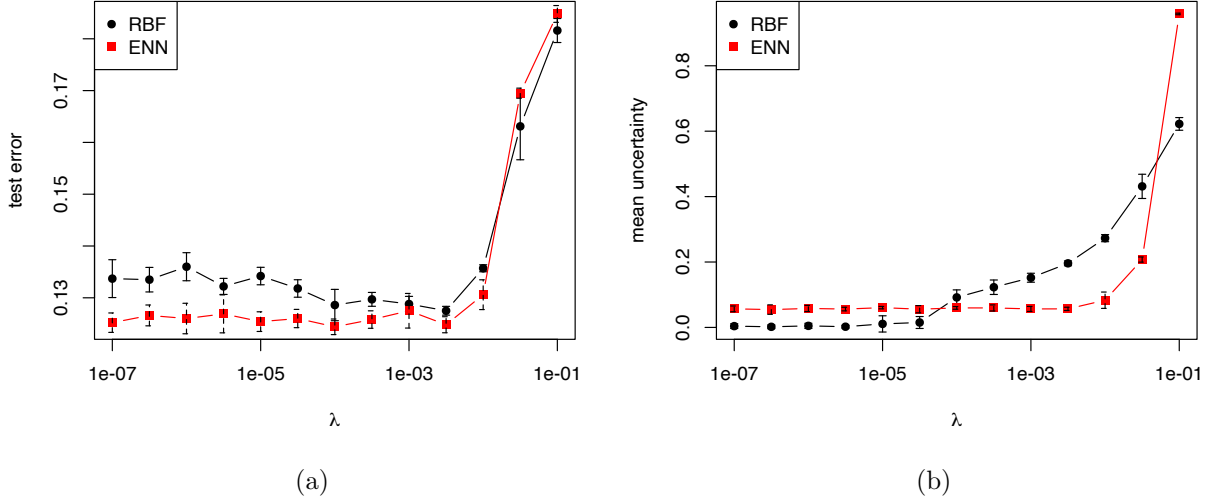


Figure 5: Test error rates (a) and mean uncertainty (b) for the ENN and RBF models, as functions of regularization parameter λ .

249 The input is the concatenated PET-CT image volume provided as a tensor of size $2 \times 256 \times$
250 256×128 , where 2 corresponds to the number of modality channels, and $256 \times 256 \times 128$
251 is the size of each input volume. The PET-CT image volumes are first fed into the feature
252 extraction module, which outputs high-level features in the form of a tensor of size $256 \times$
253 $256 \times 128 \times H$, where H is the number of features computed at each voxel. This tensor is
254 then fed into the evidential layer, which outputs mass functions representing evidence about
255 the class of each voxel, resulting in a tensor of size $256 \times 256 \times 128 \times (K + 1)$, where $K + 1$
256 is the number of masses (one for each class and one for the frame of discernment Ω). The
257 whole network is trained end-to-end by minimizing a regularized Dice loss. The different
258 components of this model are described in greater detail below.

259 *Feature extraction module.* The feature extraction module is based on a UNet [10] with
260 residual encoder and decoder layers [51], as shown in Figure 9. Each down-sampling layer
261 (marked in blue) is composed of convolution, normalization, dropout and activation blocks.
262 Each up-sampling layer (marked in green) is composed of transpose convolution, normal-
263 ization, dropout and activation blocks. The last layer (marked in yellow) is the bottom
264 connection which does not down or up-sample the data. In the experiments reported in Sec-
265 tion 5, the channels (number of filters) were set as (8, 16, 32, 64, 128) with kernel size equal
266 to 5 and convolutional strides equal to (2, 2, 2, 2). The spatial dimension, input channel and
267 output channel of the module were set, respectively, as 3, 2, and the number H of extracted
268 features. (Experiments with several values of H are reported in Section 5.2). The dropout
269 rate was set as 0 and no padding operation was applied. Instance normalization [52] was
270 used to perform intensity normalization across the width, height and depth of a single fea-

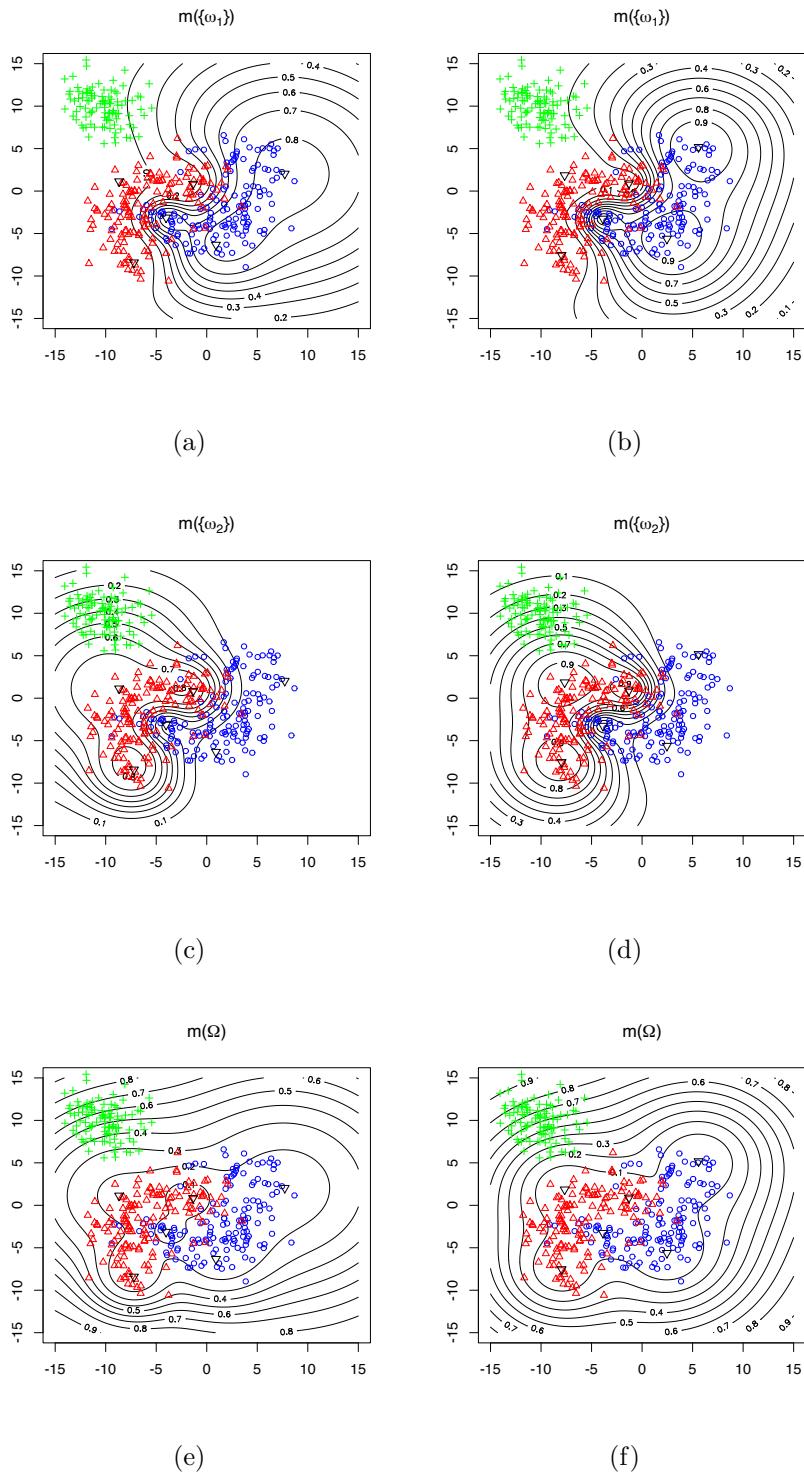


Figure 6: Contours of the mass assigned to $\{\omega_1\}$, $\{\omega_2\}$ and Ω by the RBF (left column) and ENN (right column) models. The training data are displayed in blue and red, and the third class (absent from the training data) is shown in green. Training was done with $\lambda = 0.001$ for the two models.

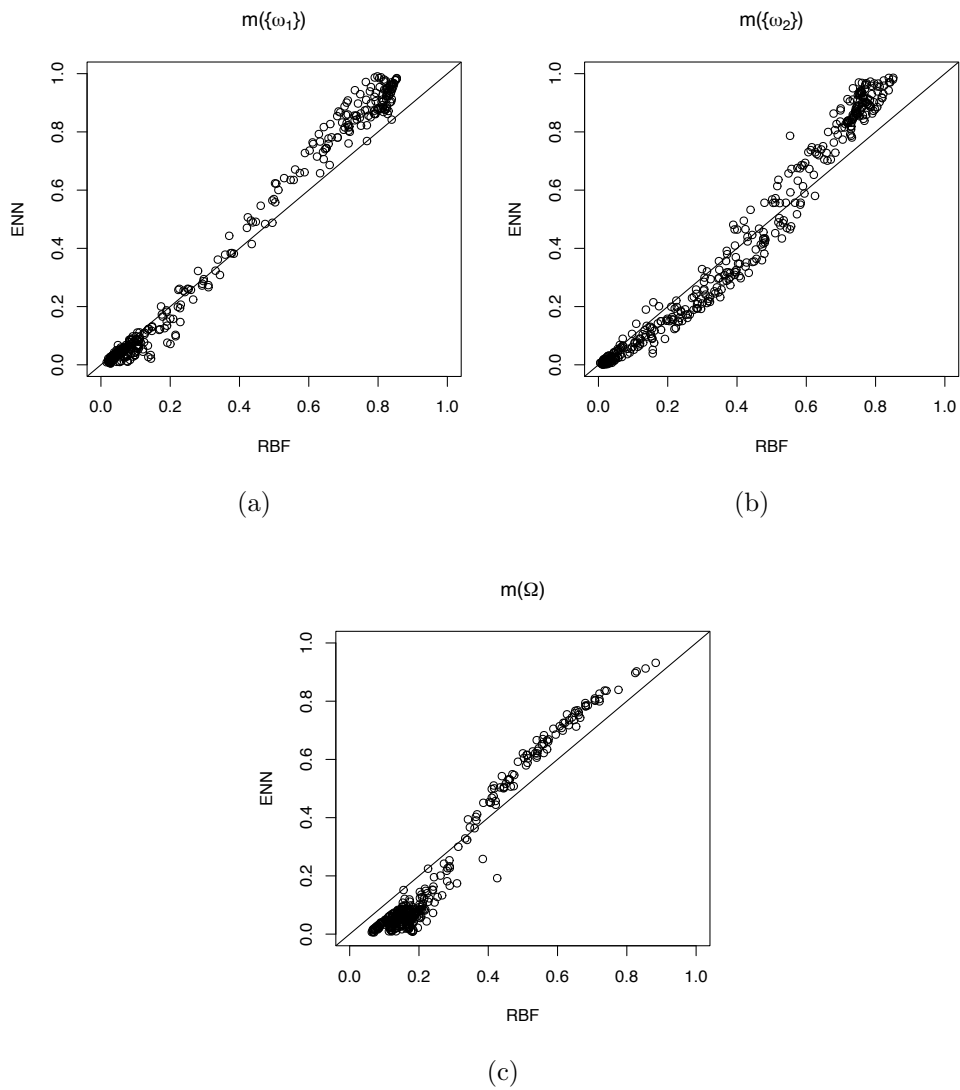


Figure 7: Masses computed by the RBF network (horizontal axis) versus the ENN model (vertical axis) for the extended dataset.

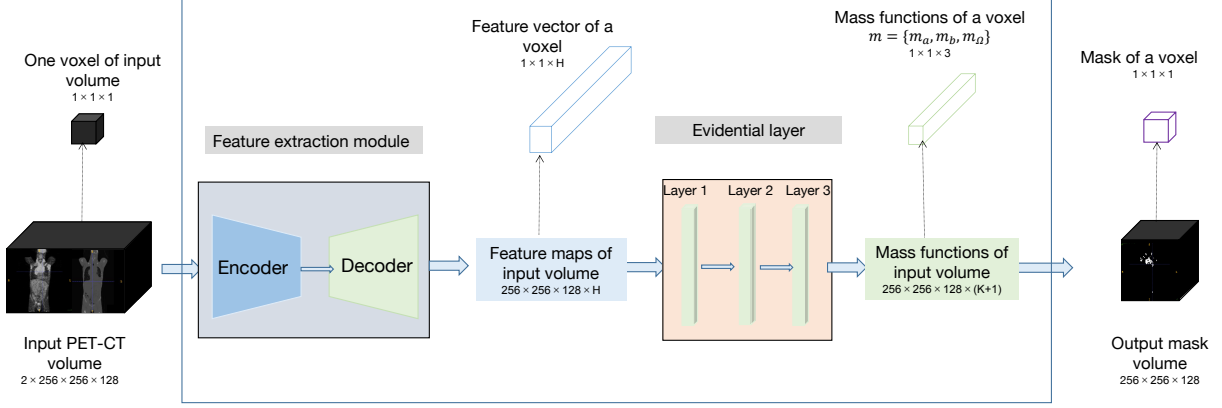


Figure 8: Global lymphoma segmentation model.

271 ture map of a single example. The Parametric Rectified Linear Unit (PReLU) function [53],
 272 which generalizes the traditional rectified unit with a slope for negative values, was used as
 273 the activation function. For each input voxel, the feature extraction module outputs a $1 \times H$
 274 feature vector, which is fed into the evidential layer.

275 *Evidential layer.* A probabilistic network with a softmax output layer may assign voxels a
 276 high probability of belonging to one class while the segmentation uncertainty is actually
 277 high because, e.g., the feature vector describing that voxel is far away from feature vectors
 278 presented during training. Here, we propose to plug-in one of the evidential classifiers
 279 described in Section 3 at the output of the feature extraction module. The ENN or RBF
 280 classifier then takes as inputs the high-level feature vectors computed by the UNet and
 281 computes, for each voxel n , a mass function m_n on the frame $\Omega = \{\omega_1, \omega_2\}$, where ω_1 and
 282 ω_2 denote, respectively, the background and the lymphoma class. We will use the names
 283 “ENN-UNet” and “RBF-UNet” to designate the two variants of the architecture.

284 *Loss function.* The whole network is trained end-to-end by minimizing a regularized Dice
 285 loss. We use the Dice loss instead of the original cross-entropy loss in UNet because the
 286 quality of the segmentation is finally assessed by the Dice coefficient. The Dice loss is defined
 287 as

$$\text{loss}_D = 1 - \frac{2 \sum_{n=1}^N S_n G_n}{\sum_{n=1}^N S_n + \sum_{n=1}^N G_n}, \quad (17)$$

288 where N is the number of voxels in the image volume, S_n is the output pignistic probability
 289 of the tumor class (i.e., $m_n(\{\omega_2\}) + m_n(\Omega)/2$) for voxel n , and G_n is ground truth for voxel
 290 n , defined as $G_n = 1$ if voxel n corresponds to a tumor, and $G_n = 0$. The regularized loss
 291 function is

$$\text{loss} = \text{loss}_D + \lambda R, \quad (18)$$

292 where λ is the regularization coefficient and R is a regularizer defined either as $R = \sum_i \alpha_i$
 293 if the ENN classifier is used in the ES module, or as $R = \sum_i v_i^2$ if the RBF classifier is used.

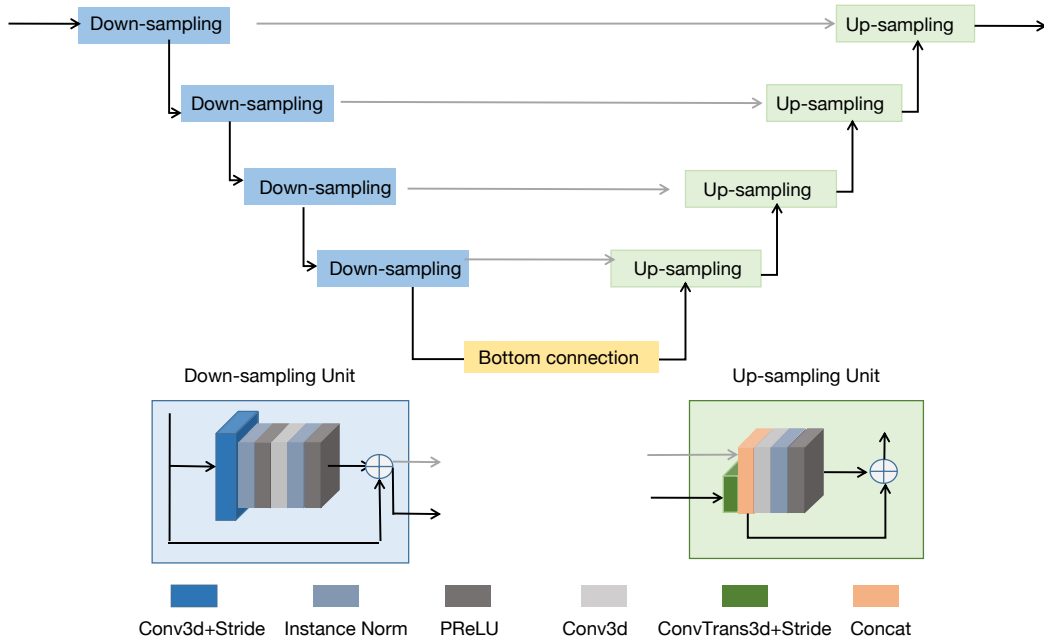


Figure 9: Feature extraction module.

294 The regularization term allows us to decrease the influence of unimportant prototypes and
 295 avoid overfitting.

296 5. Experiments

297 The model introduced in Section 4 was applied to a set of PET-CT data recorded on
 298 patients with lymphomas². The experimental settings are first described in Section 5.1. A
 299 sensitivity analysis with respect to the main hyperparameters is first reported in Section
 300 5.2. We then compare the segmentation accuracy and calibration of our models with those
 301 of state-of-the-art models in Sections 5.3 and 5.4, respectively.

302 5.1. Experimental settings

303 *Dataset.* The dataset considered in this paper contains 3D images from 173 patients who
 304 were diagnosed with large B-cell lymphomas and underwent PET-CT examination. (The
 305 study was approved as a retrospective study by the Henri Becquerel Center Institutional
 306 Review Board). The lymphomas in mask images were delineated manually by experts and
 307 considered as ground truth. All PET/CT data were stored in the DICOM (Digital Imaging
 308 and Communication in Medicine) format. The size and spatial resolution of PET and CT
 309 images and the corresponding mask images vary due to the use of different imaging machines
 310 and operations. For CT images, the size varies from $267 \times 512 \times 512$ to $478 \times 512 \times 512$.
 311 For PET images, the size varies from $276 \times 144 \times 144$ to $407 \times 256 \times 256$.

²Our code is available at <https://github.com/iWeisskohl>.

312 *Pre-processing.* Several pre-processing methods were used to process the PET/CT data. At
 313 first, the data in DICOM format were transferred into the NIFTI (Neuroimaging Informatics
 314 Technology Initiative) format for further processing. Second, the PET, CT and mask images
 315 were normalized: (1) for PET images, we applied a random intensity shift and scale of each
 316 channel with the shift value of 0 and scale value of 0.1; (2) for CT images, the shift and
 317 scale values were set to 1000 and 1/2000; (3) for mask images, the intensity value was
 318 normalized into the $[0, 1]$ interval by replacing the outside value by 1. Third, PET and
 319 CT images were resized to $256 \times 256 \times 128$ by linear interpolation, and mask images were
 320 resized to $256 \times 256 \times 128$ by nearest neighbor interpolation. Lastly, the registration of CT
 321 and PET images was performed by B-spline interpolation. All the preprocessing methods
 322 can be found in the SimpleITK [54][55] toolkit. During training, PET and CT images
 323 were concatenated as a two-channel input. We randomly selected 80% of the data for
 324 training, 10% for validation and 10% for testing. This partition was fixed and used in all
 325 the experiments reported below.

326 *Parameter initialization.* For the evidential layer module, we considered two variants based
 327 on the ENN classifier recalled in Section 3.1 on the one hand, and on an RBF network as
 328 described in Section 3.2 on the other hand. Both approaches are based on prototypes in
 329 the space of features extracted by the UNet module. When using ENN or RBF classifiers
 330 as stand-alone classifiers, prototypes are usually initialized by a clustering algorithm such
 331 as the k -means. Here, this approach is not so easy, because the whole network is trained in
 332 an end-to-end way, and the features are constructed during the training process. However,
 333 k -means initialization can still be performed by a four-step process:

- 334 1. A standard UNet architecture (with a softmax output layer) is trained end-to-end;
- 335 2. The k -means algorithm is run in the space of features extracted by the trained UNet;
- 336 3. The evidential layer is trained alone, starting from the initial prototypes computed by
 337 the k -means;
- 338 4. The whole model (feature extraction module and evidential layer) is fine-tuned by
 339 end-to-end learning with a small learning step.

340 As an alternative method, we also considered training the feature extraction module and
 341 the evidential layer simultaneously, in which case the prototypes were initialized randomly
 342 from a normal distribution with zero mean and identity covariance matrix. For the ENN
 343 module, the initial values of parameters α_i and γ_i were set, respectively, at 0.5 and 0.01, and
 344 membership degrees u_{ik} were initialized randomly by drawing uniform random numbers and
 345 normalizing. For the RBF module, the initial value of the scale parameter γ_i of RBF was
 346 set to 0.01, and the weight v_i were drawn randomly from a standard normal distribution.

347 *Learning algorithm.* Each model was trained on the learning set with 100 epochs using
 348 the Adam optimization algorithm. The initial learning rate was set to 10^{-3} . An adjusted
 349 learning rate schedule was applied by reducing the learning rate when the training loss did
 350 not decrease in 10 epochs. The model with the best performance on the validation set
 351 was saved as the final model for testing. All methods were implemented in Python with the

352 PyTorch-based medical image framework MONAI, and were trained and tested on a desktop
 353 with a 2.20GHz Intel(R) Xeon(R) CPU E5-2698 v4 and a Tesla V100-SXM2 graphics card
 354 with 32 GB GPU memory.

Evaluation criteria. The evaluation criteria most commonly used to assess the quality of medical image segmentation algorithms are the Dice score, Sensitivity and Precision. These criteria are defined as follows:

$$\text{Dice}(P, T) = \frac{2 \times TP}{FP + 2 \times TP + FN},$$

$$\text{Sensitivity}(P, T) = \frac{TP}{TP + FN},$$

$$\text{Precision}(P, T) = \frac{TP}{TP + FP},$$

355 where TP , FP , and FN denote, respectively, the numbers of true positive, false positive,
 356 false negative voxels (See Figure 10). The reported results in the following sections were
 357 obtained by calculating these three criteria for each test 3D image and then averaging over
 358 the patients. The Dice score is a global measure of segmentation performance. It is equal
 359 to twice the volume of the intersection between the predicted and actual tumor regions,
 360 divided by the sum of the volumes of these regions. Sensitivity is the proportion, among
 361 actual tumor voxels, of voxels correctly predicted as tumor. Precision is the proportion,
 362 among predicted tumor voxels, of voxels that actually belong to the tumor region; it is,
 363 thus, an estimate of the probability that the model is correct when it predicts that a voxel is
 364 in a lymphoma region. We note that neither sensitivity, nor precision are global performance
 365 criteria. We can increase sensitivity by predicting the tumor class more often (at the expense
 366 of misclassifying a lot of background pixels), and we can increase precision by being very
 367 cautious and predicting the tumor class only when it has a high probability (at the expense
 368 of missing a lot of tumor voxels). These two criteria, thus, have to be considered jointly.
 369 Finally, we can also remark that a fourth criterion can also be defined: specificity, which is
 370 the proportion, among background voxels, of voxels correctly predicted as background (i.e.,
 371 $TN/(TN + FP)$). However, as there are much more background voxels than tumor ones,
 372 this criterion is not informative in tumor segmentation applications (it is always very close
 373 to 1).

374 In addition to quality of the segmentation, we also wish to evaluate the calibration of
 375 output probabilities or belief functions (see Section 5.4). For that purpose, we will use an
 376 additional evaluation criterion, the Expected Calibration Error (ECE) [56]. The output
 377 pignistic probabilities from the evidential layer are first discretized into R equally spaced
 378 bins B_r , $r = 1, \dots, R$ (we used $R = 10$). The accuracy of bin B_r is defined as

$$\text{acc}(B_r) = \frac{1}{|B_r|} \sum_{i \in B_r} \mathbf{1}(P_i = G_i), \quad (19)$$

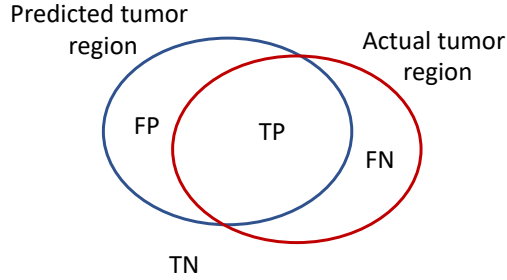


Figure 10: Geometric interpretation of the numbers of true positive (TP), false positive (FP), true negative (TN) and false negative (FN) used for the definition of evaluation criteria.

379 where P_i and G_i are, respectively, the predicted and true class labels for sample i . The
 380 average confidence of bin B_r is defined as

$$\text{conf}(B_r) = \frac{1}{|B_r|} \sum_{i \in B_r} S_i, \quad (20)$$

381 where S_i is the confidence for sample i . The ECE is the weighted average of the difference
 382 in accuracy and confidence of the bins:

$$\text{ECE} = \sum_{r=1}^R \frac{|B_r|}{N} | \text{acc}(B_r) - \text{conf}(B_r) |, \quad (21)$$

383 where N is the total number of elements in all bins, and $|B_r|$ is the number of elements in bin
 384 B_r . A model is perfectly calibrated when $\text{acc}(B_r) = \text{conf}(B_r)$ for all $r \in \{1, \dots, R\}$. Through
 385 the bin-size weighting in the ECE metric, the highly confident and accurate background
 386 voxels significantly affect the results. Because our dataset has imbalanced foreground and
 387 background proportions, we only considered voxels belonging to the tumor to calculate the
 388 ECE, similar to [57][58]. For each patient in the test set, we defined a bounding box covering
 389 the lymphoma region and calculated the ECE in this bounding box. We are interested in
 390 the patient-level ECE and thus reported the mean patient ECE instead of the voxel-level
 391 ECE (i.e., considering all voxels in the test set to calculate the ECE).

392 5.2. Sensitivity analysis

393 We analyzed the sensitivity of the results to the main design hyperparameters, which are:
 394 the number H of extracted features, the number I of prototypes and the regulation coefficient
 395 λ . The influence of the initialization method was also be studied. In all the experiments
 396 reported in this section as well as in Section 5.3, learning in each of the configurations was
 397 repeated five times with different random initial conditions.

398 *Influence of the number of features.* Table 1 shows the means and standard deviations (over
 399 five runs) of the three performance indices for ENN-UNet and RBF-UNet with different
 400 numbers of features ($H \in \{2, 5, 8\}$). The number of prototypes and the regularization
 401 coefficient were set, respectively, to $I = 10$ and $\lambda = 0$. The prototypes were initialized

Table 1: Means and standard deviations (over five runs) of the performance measures for different input dimensions H , with $I = 10$ randomly initialized prototypes and $\lambda = 0$. The best values are shown in bold.

Model	H	Dice score		Sensitivity		Precision	
		Mean	SD	Mean	SD	Mean	SD
ENN-UNet	2	0.833	0.009	0.819	0.019	0.872	0.018
	5	0.831	0.012	0.817	0.016	0.870	0.011
	8	0.829	0.006	0.816	0.010	0.877	0.019
RBF-UNet	2	0.824	0.009	0.832	0.008	0.845	0.016
	5	0.825	0.006	0.817	0.016	0.862	0.010
	8	0.821	0.011	0.813	0.010	0.862	0.022

Table 2: Means and standard deviations (over five runs) of the performance measures for different values of the regularization coefficient λ , with $I = 10$ randomly initialized prototypes and $H = 2$ features. The best values are shown in bold.

Model	λ	Dice score		Sensitivity		Precision	
		Mean	SD	Mean	SD	Mean	SD
ENN-UNet	0	0.833	0.009	0.819	0.019	0.872	0.018
	1e-4	0.822	0.007	0.818	0.026	0.839	0.035
	1e-2	0.823	0.004	0.817	0.023	0.856	0.023
RBF-UNet	0	0.824	0.009	0.832	0.008	0.845	0.016
	1e-4	0.825	0.011	0.811	0.022	0.869	0.020
	1e-2	0.829	0.010	0.818	0.022	0.867	0.016

402 randomly. ENN-UNet achieves the highest Dice score and sensitivity with $H = 2$ features,
403 but the highest precision with $H = 8$. However, the differences are small and concern only
404 the third decimal point. Similarly, RBF-UNet had the best values of the Dice score and
405 precision for $H = 5$ features, but again the differences are small. Overall, it seems that only
406 two features are sufficient to discriminate between tumor and background voxels.

407 *Influence of the regularization coefficient.* In the previous experiment, the networks were
408 trained without regularization. Tables 2 and 3 show the performances of ENN-UNet and
409 RBF-UNet for different values of λ , with $I = 10$ randomly initialized prototypes and, re-
410 spectively, $H = 2$ and $H = 8$ inputs. With both settings, ENN-UNet does not benefit from
411 regularization (the best results are obtained with $\lambda = 0$). In contrast, RBF-UNet is more
412 sensitive to regularization, and achieves the highest Dice score with $\lambda = 0.01$. This find-
413 ing confirms the remark already made in Section 3.3, where it was observed that an ENN
414 classifier seems to be less sensitive to regularization than an RBF classifier (see Figure 5a).

415 *Influence of the number of prototypes.* The number I of prototypes is another hyperpa-
416 rameter that may impact segmentation performance. Table 4 shows the performances of
417 ENN-UNet and RBF-UNet with 10 and 20 randomly initialized prototypes, the other hy-
418 perparameters being fixed at $H = 2$ and $\lambda = 0$. Increasing the number of prototypes beyond
419 10 does not seem to improve the performance of ENN-UNet, while it does slightly improve

Table 3: Means and standard deviations (over five runs) of the performance measures for different values of the regularization coefficient λ , with $I = 10$ randomly initialized prototypes and $H = 8$ features. The best values are shown in bold.

Model	λ	Dice score		Sensitivity		Precision	
		Mean	SD	Mean	SD	Mean	SD
ENN-UNet	0	0.829	0.006	0.811	0.010	0.877	0.019
	1e-4	0.827	0.008	0.809	0.019	0.873	0.024
	1e-2	0.822	0.009	0.807	0.021	0.867	0.011
RBF-UNet	0	0.821	0.010	0.813	0.010	0.862	0.022
	1e-4	0.827	0.004	0.830	0.005	0.852	0.012
	1e-2	0.832	0.006	0.825	0.022	0.867	0.020

Table 4: Means and standard deviations (over five runs) of the performance measures for different numbers I of randomly initialized prototypes, with $H = 2$ features and $\lambda = 0$. The best values are shown in bold.

Model	I	Dice score		Sensitivity		Precision	
		Mean	SD	Mean	SD	Mean	SD
ENN-UNet	10	0.833	0.009	0.819	0.019	0.872	0.018
	20	0.823	0.007	0.804	0.006	0.864	0.012
RBF-UNet	10	0.824	0.009	0.832	0.008	0.845	0.016
	20	0.830	0.007	0.810	0.012	0.867	0.010

420 the performance of RBF-UNet in terms of Dice score and precision, at the expense of an
421 increased computing time.

422 *Influence of the prototype initialization method.* Finally, we compared the two initialization
423 methods mentioned in Section 5.1. For k -means initialization, in the first step, a UNet model
424 was trained with the following settings: kernel size=5, channels =(8, 16, 32, 64, 128) and
425 strides=(2, 2, 2, 2). The spatial dimension, input and output channel were set, respectively,
426 3, 2, and 2. This pre-trained UNet was used to extract $H = 2$ features, and 10 prototypes
427 were obtained by running the k -means algorithm in the space of extracted features. These
428 prototypes were fed into ENN or RBF layers, which were trained separately, with fixed
429 features. For this step, the learning rate was set to 10^{-2} . Finally, the whole model was fine-
430 tuned end-to-end, with a smaller learning rate equal to 10^{-4} . Table 5 shows the performances
431 of ENN-UNet and RBF-UNet with random and k -means initialization. Both ENN-UNet and
432 RBF-UNet achieve a higher Dice score when using the k -means initialization method, and
433 the variability of the results is also reduced with this method.

434 Not only does the k -means initialization method slightly improve the performances of
435 ENN-UNet and RBF-UNet quantitatively, but it also tends to position the prototypes in
436 regions of high data density. As a result, a high output mass $m(\Omega)$ signals that the input
437 data is atypical. In that sense, the output mass function is more interpretable. This point
438 is illustrated by Figures 11 and 12, which show the contours, in the two-dimensional feature
439 space, of the masses assigned to the background, the tumor class and the frame of discern-

Table 5: Means and standard deviations (over five runs) of the performance measures for different initialization methods, with $I = 10$ prototypes, $H = 2$ features and $\lambda = 0$. The best values are shown in bold.

Model	Initialization	Dice score		Sensitivity		Precision	
		Mean	SD	Mean	SD	Mean	SD
ENN-UNet	Random	0.833	0.009	0.819	0.019	0.872	0.018
	k -means	0.846	0.002	0.830	0.004	0.879	0.008
RBF-UNet	Random	0.824	0.009	0.832	0.008	0.845	0.016
	k -means	0.839	0.003	0.824	0.001	0.879	0.008

440 ment when using k -means initialization (with $\lambda = 10^{-2}$ and $I = 10$) with, respectively,
 441 ENN-UNet and RBF-UNet. For both models, the prototypes are well distributed over the
 442 two classes, and the mass on Ω decreases with the distance to the data, as expected. In
 443 contrast, when using random initialization (as shown in Figure 13 for the ENN-UNet model
 444 – results are similar with the RBF-UNet model), the prototypes are located in the back-
 445 ground region, and the mass $m(\Omega)$ does not have a clear meaning (although the decision
 446 boundary still ensures a good discrimination between the two classes).

447 From this sensitivity analysis, we can conclude that the performances of both ENN-
 448 UNet and RBF-UNet are quite robust to the values of the hyperparameters, and that the
 449 two models achieve comparable performances. The k -means initialization method seems to
 450 yield better results, both quantitatively and qualitatively. The next section is devoted to a
 451 comparison with alternative models.

452 5.3. Comparative analysis: segmentation accuracy

453 In this section, we compare the performances of the ENN-UNet and RBF-UNet models
 454 with those of the baseline model, UNet [10], as well as three state-of-the-art models reviewed
 455 in Section 1: VNet [11], SegResNet [12] and nnUNet [13]. For all compared methods, the
 456 same learning set and pre-processing steps were used. All the compared methods were
 457 trained with the Dice loss function (17). Details about the optimization algorithm were
 458 given in Section 5.1. All methods were implemented based on the MONAI framework³
 459 and can be called directly. For UNet, the kernel size was set as 5 and the channels were
 460 set to (8, 16, 32, 64, 128) with strides=(2, 2, 2, 2). For nnUNet, the kernel size was set as
 461 (3, (1, 1, 3), 3, 3) and the upsample kernel size was set as (2, 2, 1) with strides ((1, 1, 1), 2, 2, 1).
 462 For SegResNet [12] and VNet [11], we used the pre-defined model without changing any
 463 parameter. The spatial dimension, input channel and output channel were set, respectively,
 464 3, 2, and 2 for the four compared models. As for other hyperparameters not mentioned
 465 here, we used the pre-defined value given in MONAI. As shown by the sensitivity analysis
 466 performed in Section 5.2, the best results for ENN-UNet and RBF-UNet are achieved with
 467 $\lambda = 0$, $I = 10$, $H = 2$ and k -means initialization.

³More details about how to use those models can be found from MONAI core tutorials <https://monai.io/started.html#monaicore>.

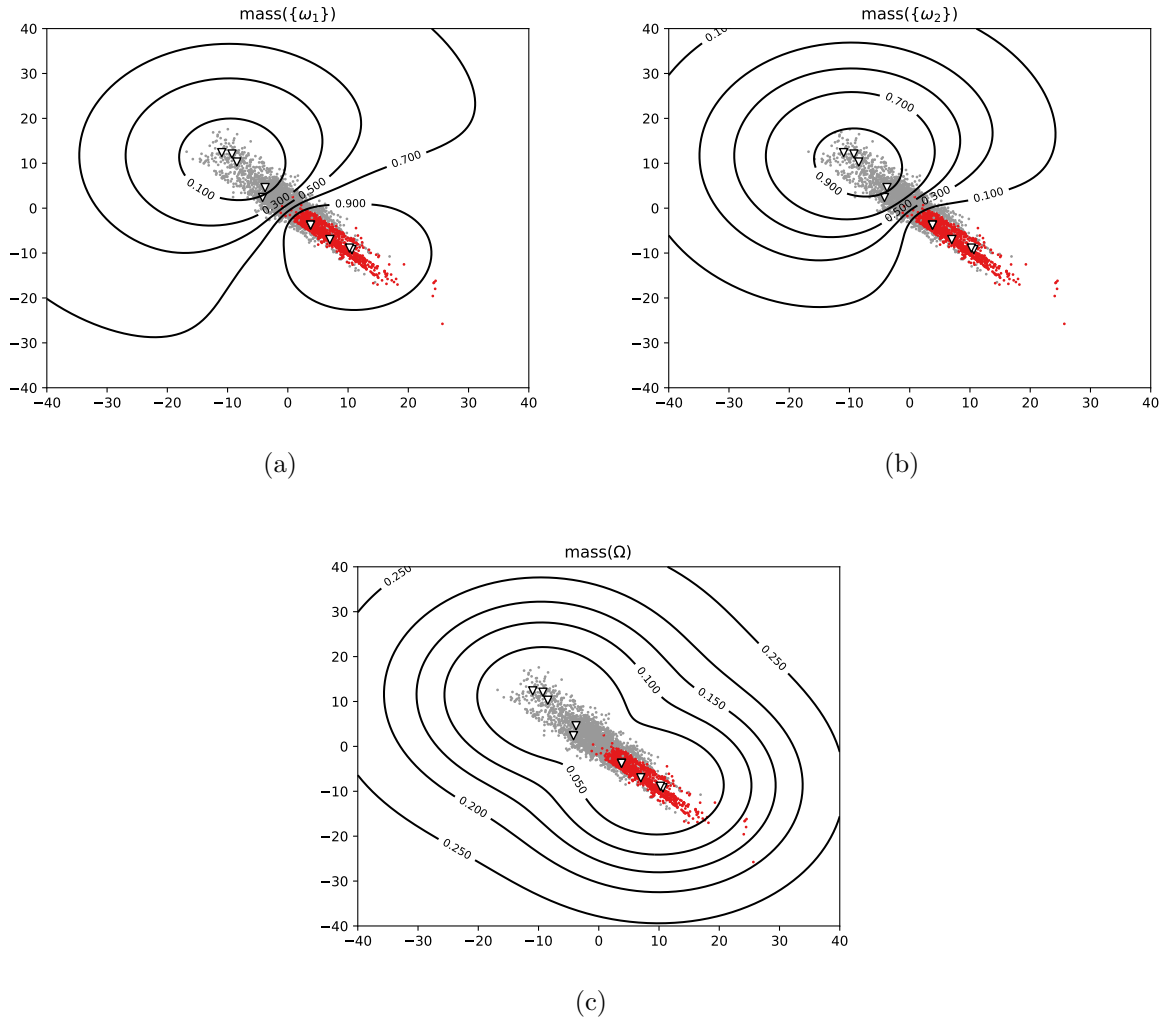


Figure 11: Contours in feature space of the masses assigned to the background (a), the tumor class (b) and the frame of discernment (c) by the ENN-UNet model initialized by k -means. Training was done with $\lambda = 10^{-2}$, $H = 2$ and $I = 10$. Sampled feature vectors from the tumor and background classes are marked in gray and red, respectively.

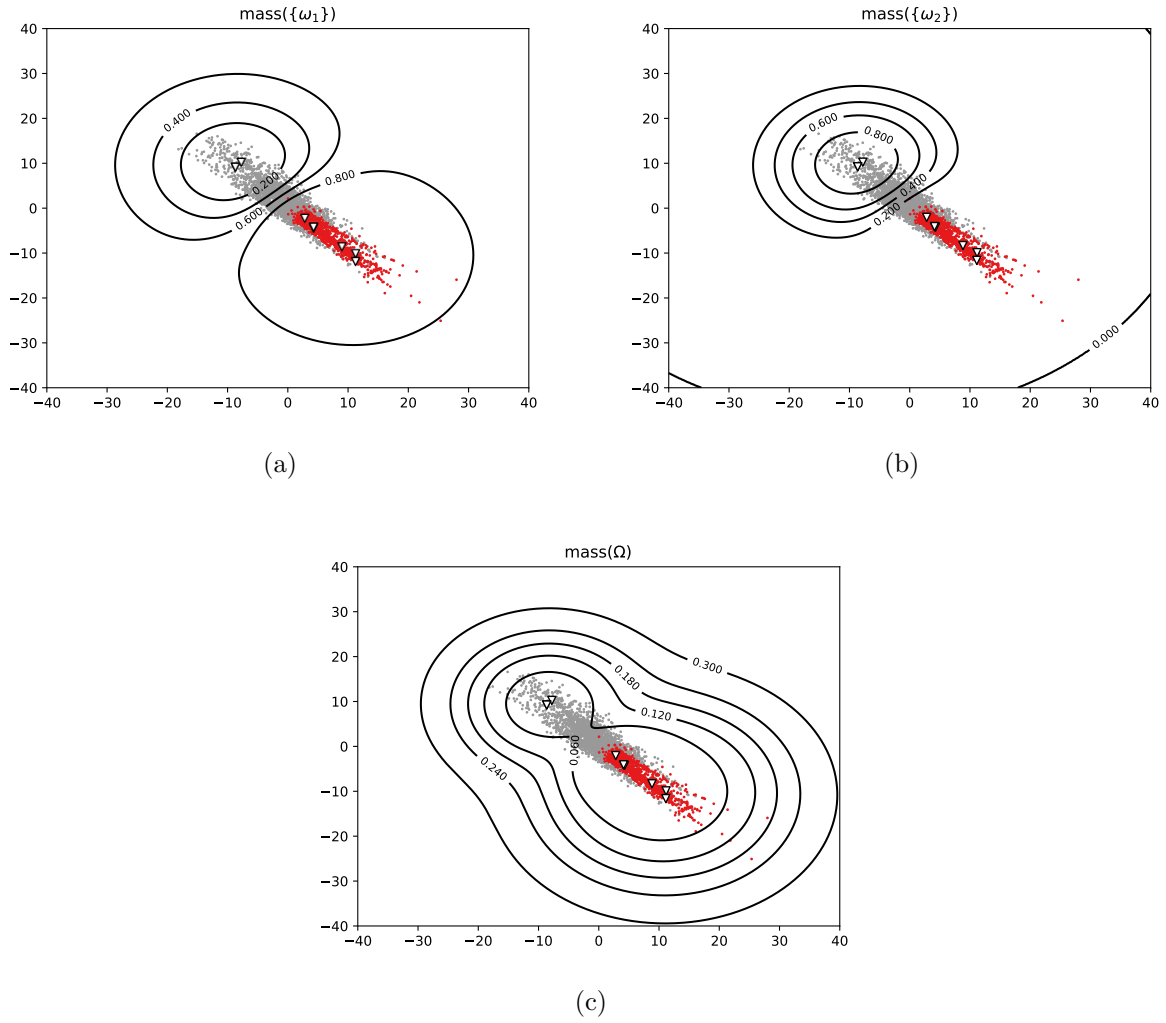
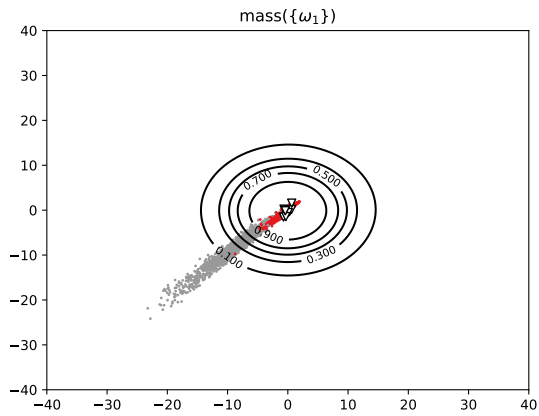
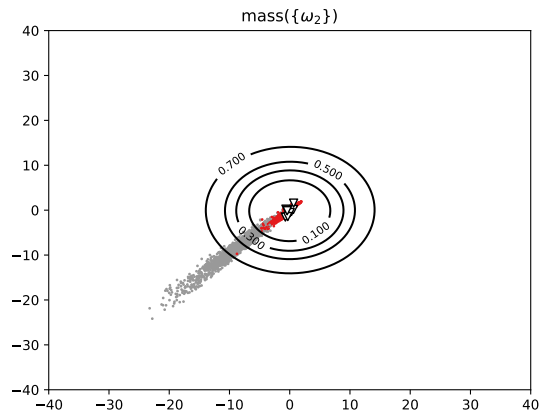


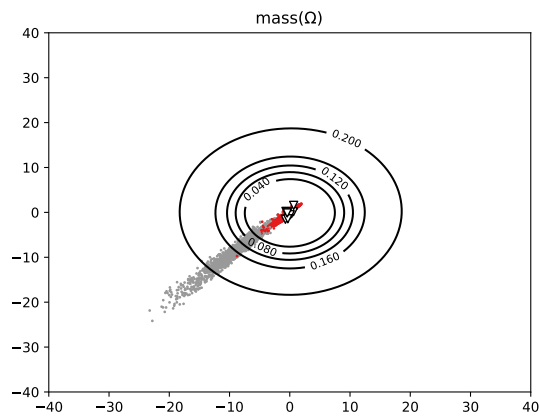
Figure 12: Contours in feature space of the masses assigned to the background (a), the tumor class (b) and the frame of discernment (c) by the RBF-UNet model initialized by k -means. Training was done with $\lambda = 10^{-2}$, $H = 2$ and $I = 10$. Sampled feature vectors from the tumor and background classes are marked in gray and red, respectively.



(a)



(b)



(c)

Figure 13: Contours in feature space of the masses assigned to the background (a), the tumor class (b) and the frame of discernment (c) by the ENN-UNet model initialized randomly. Training was done with $\lambda = 10^{-2}$, $H = 2$ and $I = 10$. Sampled feature vectors from the tumor and background classes are marked in gray and red, respectively.

Table 6: Means and standard deviations (over five runs) of the performance measures for ENN-UNet, RBF-UNet and four reference methods. The best result is shown in bold, and the second best is underlined.

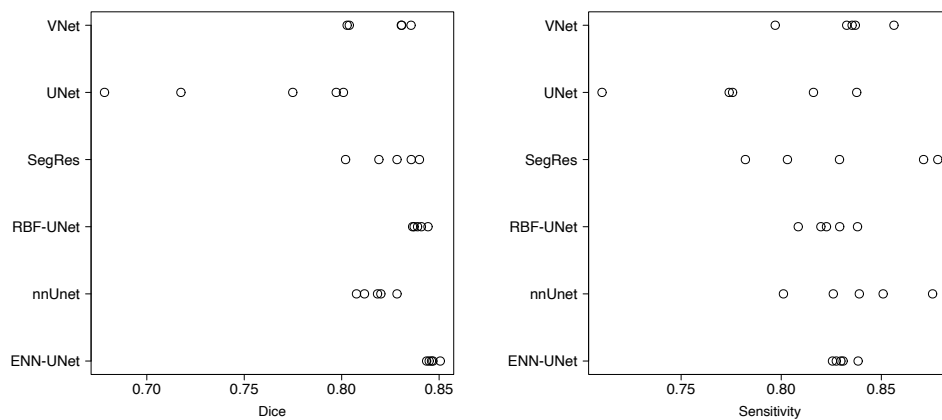
Model	Dice score		Sensitivity		Precision	
	Mean	SD	Mean	SD	Mean	SD
UNet [51]	0.753	0.054	0.782	0.048	<u>0.896</u>	0.047
nnUNet [13]	0.817	0.008	0.838	0.028	0.879	0.032
VNet [11]	0.820	0.016	0.831	0.021	0.901	0.056
SegResNet [12]	0.825	0.015	<u>0.832</u>	0.042	0.876	0.051
ENN-UNet	0.846	0.002	0.830	0.004	0.879	0.008
RBF-UNet	<u>0.839</u>	0.003	0.824	0.001	0.879	0.008

Table 7: Conover-Iman test of multiple comparisons between the Dice scores obtained by the six models: t-test statistics and p-values. P-values less than 0.01 are printed in bold.

	ENN-UNet	nnUNet	RBF-UNet	SegResNet	UNet
nnUNet	6.759				
	0.0000				
RBF-UNet	2.156	-4.602			
	0.0857	0.0004			
SegResNet	5.349	-1.410	3.193		
	0.0001	0.3282	0.0088		
UNet	10.283	3.524	8.127	4.934	
	0.0000	0.0043	0.0000	0.0002	
VNet	6.054	-0.705	3.898	0.705	-4.229
	0.0000	0.8091	0.0019	0.8669	0.0009

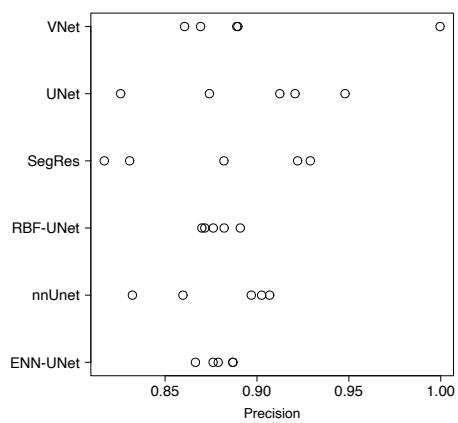
468 The means and standard deviations of the Dice score, sensitivity and precision over five
469 runs with random initialization for the six methods are shown in Table 6, and the raw values
470 are plotted in Figure 14. We can see that ENN-UNet and RBF-UNet achieve, respectively,
471 the highest and the second highest mean Dice score. A Kruskal-Wallis test performed on
472 the whole data concludes to a significant difference between the distributions of the Dice
473 score for the six methods (p-value = 0.0001743), while the differences are not significant for
474 sensitivity (p-value = 0.2644) and precision (p-value = 0.9496). Table 7 shows the results of
475 the Conover-Iman test of multiple comparisons [59][60] with Benjamini-Yekutieli adjustment
476 [61]. We can see that the differences between the Dice scores obtained by ENN-UNet and
477 RBF-UNet on the one hand, and the four other methods on the other hand are highly
478 significant (p-values $< 10^{-2}$), while the difference between ENN-UNet and RBF-UNet is
479 only weakly significant (p-value = 0.0857).

480 Figure 15 shows two examples of segmentation results obtained by ENN-UNet and UNet,
481 corresponding to large and isolated lymphomas. We can see, in these two examples, that
482 UNet is more conservative (it correctly detects only a subset of the tumor voxels), which
483 may explain why it has a relatively high precision. However, the tumor regions predicted



(a)

(b)



(c)

Figure 14: Values of the Dice score (a), sensitivity (b) and precision (c) for five runs of the six methods.

484 by ENN-UNet better overlap the ground-truth tumor region, which is also reflected by the
485 higher Dice score.

486 5.4. Comparative analysis: calibration

487 Besides segmentation accuracy, another important issue concerns the quality of uncer-
488 tainty quantification. Monte-Carlo dropout (MCD) [25] is a state-of-the-art technique
489 for improving uncertainty quantification capabilities of deep networks. In this section, we
490 compare the ECE (21) achieved by UNet (the baseline), SegResNet (the best alternative
491 method found in Section 5.3), and our proposals: ENN-UNet, and RBF-UNet, with and
492 without MCD. For the four methods, the dropout rate was set to 0.5 and the sample num-
493 ber was set to 20; we averaged the 20 output probabilities (the pignistic probabilities for the
494 two evidential models) at each voxel as the final output of the model.

495 The results are reported in Table 8. We can see that MCD enhances the segmentation
496 performance (measured by the Dice index) of UNet et SegResNet, and improves the cali-
497 bration of all methods, except SegResNet. Overall, the smallest average ECE is achieved
498 by RBF-UNet and ENN-UNet with MCD, but the standard deviations are quite large. A
499 Kruskal-Wallis test concludes to a significant difference between the distributions of ECE
500 for the eight methods (p-value = 0.01). The p-values of the Conover-Iman test of multi-
501 ple comparisons with Benjamini-Yekutieli adjustment reported in Table 9 show significant
502 differences between the ECE of RBF-UNet with MCD on the one hand, and those of RBF-
503 UNet without MCD, SegResNet with MCD, and UNet without MCD on the other hand.
504 We also tested the pairwise differences between the ECE values obtained by RBF-UNet and
505 ENN-UNet with MCD on the one hand, and UNet with and without MCD as well as Seg-
506 ResNet with and without MCD on the other hand using the Wilcoxon rank sum test. The
507 corresponding p-values are shown in Table 10. We find significant differences between the
508 ECE RBF-UNet with MCD and those of the other methods, but only a weakly significant
509 difference between ENN-UNet with MCD and UNet without MCD. In summary, there is
510 some evidence that MCD improves calibration, even for evidential models, and that the best
511 calibration is achieved by the RBF-UNet model, but this evidence is not fully conclusive
512 due to the limited size of the dataset; our findings will have to be confirmed by further
513 experiments with larger datasets.

514 6. Conclusion

515 An evidential framework for segmenting lymphomas from 3D PET-CT images with un-
516 certainty quantification has been proposed in this paper. Our architecture is based on the
517 concatenation of a UNet, which extracts high-level features from the input images, and
518 an evidential segmentation module, which computes output mass functions for each voxel.
519 Two versions of this evidential module, both involving prototypes, have been studied: one
520 is based on the ENN model initially proposed as a stand-alone classifier in [31], while the
521 other one relies on an RBF layer and the addition of weight of evidence. The whole model
522 is trained end-to-end by minimizing the Dice loss. The initialization of prototypes has been
523 shown to be a crucial step in this approach. The best method found has been to pre-train

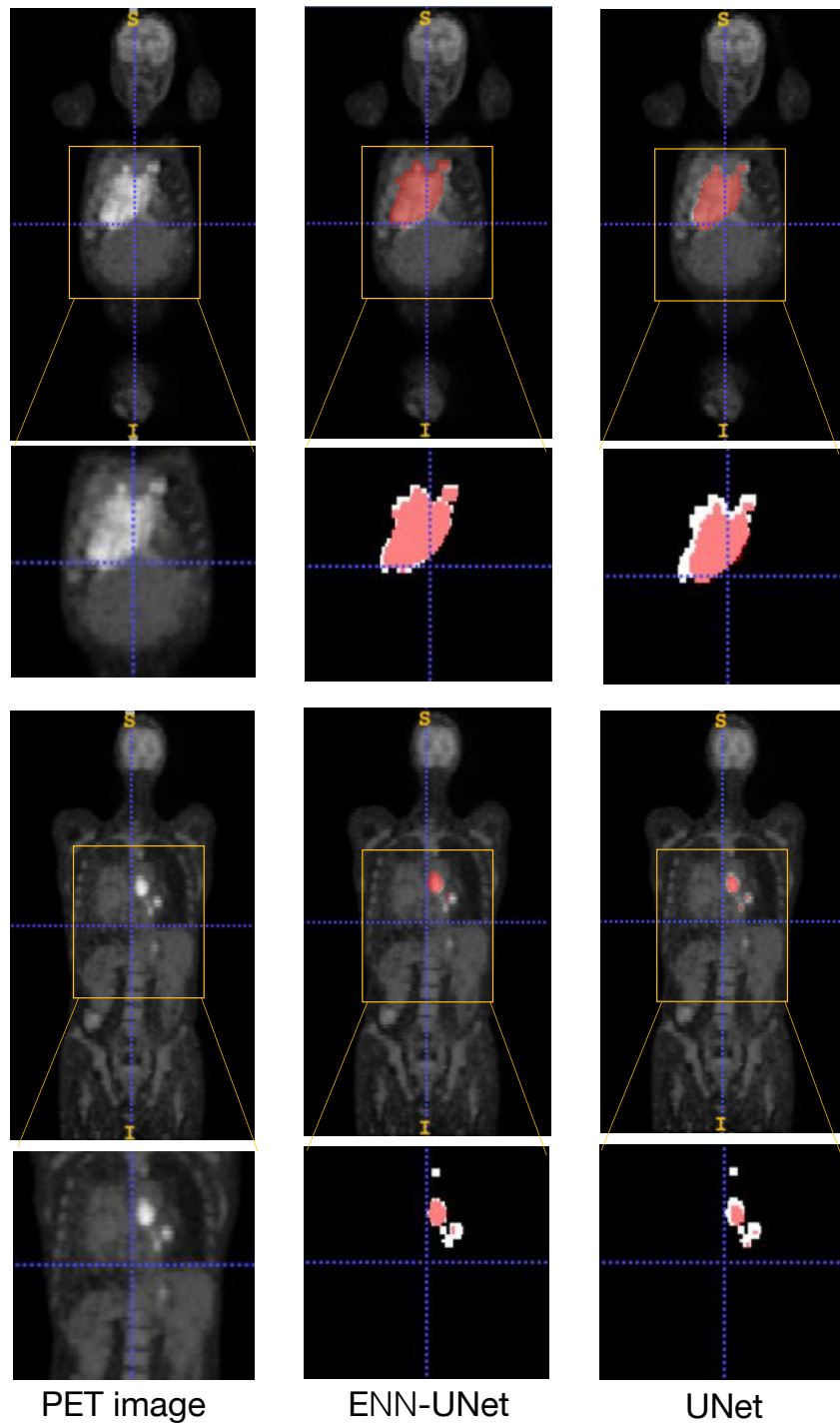


Figure 15: Two examples of segmentation results by ENN-UNet and UNet. The first and the second row are, respectively, representative of large and isolated small lymphomas. The three columns correspond, from left to right, to the PET images and the segmentation results obtained by ENN-UNet and UNet. The white and red region represent, respectively, the ground truth and the segmentation result.

Table 8: Means and standard deviations (over five runs) of the Dice score and ECE for UNet, SegResNet, ENN-UNet and RBF-UNet, with and without MCD. The best results are shown in bold, the second best are underlined.

Model	Dice score		ECE(%)	
	Mean	SD	Mean	SD
UNet	0.754	0.054	2.22	0.205
SegResNet	0.825	0.015	1.97	0.488
ENN-UNet	0.846	0.002	1.99	0.110
RBF-UNet	0.839	0.003	2.12	0.028
UNet with MC	0.828	0.005	1.93	0.337
SegResNet with MC	<u>0.844</u>	0.009	2.53	0.973
ENN-UNet with MC	0.841	0.003	<u>1.53</u>	0.075
RBF-UNet with MC	0.840	0.003	1.52	0.041

Table 9: Conover-Iman test of multiple comparisons between the ECE obtained by UNet, SegResNet, ENN and RBF, with and without MCD: t-test statistics and p-values. P-values less than 0.01 are printed in bold.

	ENN	ENN-MC	RBF	RBF-MC	SegRes	SegRes-MC	UNet
ENN-MC	0.926						
	1.0000						
RBF	-1.191	-2.118					
	0.7403	0.2892					
RBF-MC	2.812	1.886	4.004				
	0.1145	0.3419	0.0095				
SegRes	0.695	-0.232	1.886	-2.117			
	1.0000	1.0000	0.3761	0.3305			
SegRes-MC	-0.860	-1.787	0.331	-3.673	-1.555		
	1.0000	0.3530	1.0000	0.0159	0.4756		
UNet	-1.357	-2.283	-0.165	-4.169	-2.051	-0.496	
	0.6337	0.2677	1.0000	0.0119	0.2962	1.0000	
UNet-MC	0.430	-0.496	1.621	-2.382	-0.265	1.290	1.787
	1.0000	1.0000	0.4507	0.2564	1.0000	0.6667	0.3824

Table 10: P-values for the Wilcoxon rank sum test applied to the comparison of ECE obtained by ENN-UNet and RBF UNet with MCD on the one hand, and the four other methods on the other hand (UNet and SegResNet with and without MCD).

	UNet	UNet-MC	SegRes	SegRes-MC
ENN-MC	0.095	0.67	0.69	0.31
RBF-MC	0.0079	0.012	0.055	0.0079

524 a UNet with a softmax output layer, initialize the prototype with the k -means algorithm
525 in the space of extracted features, train the evidential layer separately, and fine-tune the
526 whole network. Our model has been shown to outperform the baseline UNet model as well
527 as other state-of-the-art segmentation method on a dataset of 173 patients with lymphomas.
528 Preliminary results also suggest the outputs of the evidential models (in particular, the one
529 with an RBF layer) are better calibrated and that calibration error can be further decreased
530 by Monte Carlo dropout. These results, however, will have to be confirmed by further
531 experiments with larger datasets.

532 This work can be extended in many directions. One of them is to further evaluate
533 the approach by applying it to other medical image segmentation problems. One of the
534 potential problems that may arise is related to the dimensionality of the feature space. In
535 the application considered in this paper, good results were obtained with only two extracted
536 features. If some other learning tasks require a much larger number of features, we may need
537 a much higher number of prototypes and learning may be slow. This issue could be addressed
538 by adapting the loss function as proposed, e.g., in [62]. We also plan to further study the
539 calibration properties of the belief functions computed by our approach (using calibration
540 measures specially designed for belief functions), as well as the novelty detection capability
541 of our model.

542 Acknowledgements

543 This work was supported by the China Scholarship Council (No. 201808331005). It
544 was carried out in the framework of the Labex MS2T, which was funded by the French
545 Government, through the program “Investments for the future” managed by the National
546 Agency for Research (Reference ANR-11-IDEX-0004-02)

547 References

- 548 [1] Y. S. Jhanwar, D. J. Straus, The role of PET in lymphoma, *Journal of Nuclear Medicine* 47 (8) (2006)
549 1326–1334.
- 550 [2] H. Zaidi, I. El Naqa, PET-guided delineation of radiation therapy treatment volumes: a survey of image
551 segmentation techniques, *European Journal of Nuclear Medicine and Molecular Imaging* 37 (11) (2010)
552 2165–2187.
- 553 [3] H. Ilyas, N. G. Mikhaeel, J. T. Dunn, F. Rahman, H. Møller, D. Smith, S. F. Barrington, Defining
554 the optimal method for measuring baseline metabolic tumour volume in diffuse large B cell lymphoma,
555 *European journal of nuclear medicine and molecular imaging* 45 (7) (2018) 1142–1154.
- 556 [4] F. Eude, M. N. Toledano, P. Vera, H. Tilly, S.-D. Mihailescu, S. Becker, Reproducibility of baseline
557 tumour metabolic volume measurements in diffuse large B-cell lymphoma: Is there a superior method?,
558 *Metabolites* 11 (2) (2021) 72.
- 559 [5] D. Onoma, S. Ruan, S. Thureau, et al., Segmentation of heterogeneous or small FDG PET positive
560 tissue based on a 3d-locally adaptive random walk algorithm, *Computerized Medical Imaging and
561 Graphics* 38 (8) (2014) 753–763.
- 562 [6] H. Hu, P. Decazes, P. Vera, H. Li, S. Ruan, Detection and segmentation of lymphomas in 3D PET images
563 via clustering with entropy-based optimization strategy, *International journal of computer assisted
564 radiology and surgery* 14 (10) (2019) 1715–1724.
- 565 [7] H. Li, H. Jiang, S. Li, et al., DenseX-net: an end-to-end model for lymphoma segmentation in whole-
566 body PET/CT images, *IEEE Access* 8 (2019) 8004–8018.

- 567 [8] H. Hu, L. Shen, T. Zhou, et al., Lymphoma segmentation in PET images based on multi-view and
568 conv3d fusion strategy, in: 2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI),
569 IEEE, 2020, pp. 1197–1200.
- 570 [9] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Pro-
571 ceedings of the IEEE conference on computer vision and pattern recognition, Boston, USA, 2015, pp.
572 3431–3440.
- 573 [10] O. Ronneberger, P. Fischer, T. Brox, U-Net: convolutional networks for biomedical image segmentation,
574 in: N. Navab, J. Hornegger, W. M. Wells, A. F. Frangi (Eds.), Medical Image Computing and Computer-
575 Assisted Intervention – MICCAI 2015, Springer International Publishing, Cham, 2015, pp. 234–241.
- 576 [11] F. Milletari, N. Navab, S.-A. Ahmadi, V-net: Fully convolutional neural networks for volumetric medical
577 image segmentation, in: 2016 fourth international conference on 3D vision, IEEE, 2016, pp. 565–571.
- 578 [12] A. Myronenko, 3D MRI brain tumor segmentation using autoencoder regularization, in: International
579 MICCAI Brain lesion Workshop, Springer, 2018, pp. 311–320.
- 580 [13] F. Isensee, J. Petersen, A. Klein, D. Zimmerer, et al., Nnu-net: Self-adapting framework for u-net-based
581 medical image segmentation, arXiv preprint arXiv:1809.10486.
- 582 [14] P. Blanc-Durand, S. Jégou, S. Kanoun, et al., Fully automatic segmentation of diffuse large B cell lym-
583 phoma lesions on 3D FDG-PET/CT for total metabolic tumour volume prediction using a convolutional
584 neural network., European Journal of Nuclear Medicine and Molecular Imaging (2020) 1–9.
- 585 [15] L. Huang, T. Dencœux, D. Tonnelet, P. Decazes, S. Ruan, Deep pet/ct fusion with dempster-shafer
586 theory for lymphoma segmentation, in: C. Lian, X. Cao, I. Rekik, X. Xu, P. Yan (Eds.), Machine
587 Learning in Medical Imaging, Springer International Publishing, Cham, 2021, pp. 30–39.
- 588 [16] G. Shafer, A mathematical theory of evidence, Vol. 42, Princeton University Press, 1976.
- 589 [17] E. Hüllermeier, W. Waegeman, Aleatoric and epistemic uncertainty in machine learning: An introduc-
590 tion to concepts and methods, Machine Learning 110 (3) (2021) 457–506.
- 591 [18] J. Quiñonero-Candela, M. Sugiyama, N. D. Lawrence, A. Schwaighofer, Dataset shift in machine learn-
592 ing, MIT Press, 2009.
- 593 [19] R. Mehta, T. Christinck, T. Nair, P. Lemaitre, D. Arnold, T. Arbel, Propagating uncertainty across
594 cascaded medical imaging tasks for improved deep learning inference, in: Uncertainty for Safe Utiliza-
595 tion of Machine Learning in Medical Imaging and Clinical Image-Based Procedures, Springer, 2019,
596 pp. 23–32.
- 597 [20] W. J. Maddox, P. Izmailov, T. Garipov, D. P. Vetrov, A. G. Wilson, A simple baseline for Bayesian
598 uncertainty in deep learning, Advances in Neural Information Processing Systems 32 (2019) 13153–
599 13164.
- 600 [21] L. Yu, S. Wang, X. Li, C.-W. Fu, P.-A. Heng, Uncertainty-aware self-ensembling model for semi-
601 supervised 3d left atrium segmentation, in: International Conference on Medical Image Computing
602 and Computer-Assisted Intervention, Springer, 2019, pp. 605–613.
- 603 [22] F. C. Ghesu, B. Georgescu, A. Mansoor, Y. Yoo, E. Gibson, R. Vishwanath, A. Balachandran, J. M.
604 Balter, Y. Cao, R. Singh, et al., Quantifying and leveraging predictive uncertainty for medical image
605 assessment, Medical Image Analysis 68 (2021) 101855.
- 606 [23] G. E. Hinton, D. van Camp, Keeping the neural networks simple by minimizing the description length
607 of the weights, in: Proceedings of the Sixth Annual Conference on Computational Learning Theory,
608 COLT '93, Association for Computing Machinery, New York, NY, USA, 1993, p. 5?13.
- 609 [24] D. J. MacKay, A practical Bayesian framework for backpropagation networks, Neural computation
610 4 (3) (1992) 448–472.
- 611 [25] Y. Gal, Z. Ghahramani, Dropout as a Bayesian approximation: Representing model uncertainty in
612 deep learning, in: International Conference on Machine Learning, PMLR, 2016, pp. 1050–1059.
- 613 [26] D. Tran, M. W. Dusenberry, M. van der Wilk, D. Hafner, Bayesian layers: A module for neural network
614 uncertainty, arXiv preprint arXiv:1812.03973.
- 615 [27] A. P. Dempster, Upper and lower probability inferences based on a sample from a finite univariate
616 population, Biometrika 54 (3-4) (1967) 515–528.
- 617 [28] T. Dencœux, D. Dubois, H. Prade, Representations of uncertainty in artificial intelligence: Beyond

- 618 probability and possibility, in: P. Marquis, O. Papini, H. Prade (Eds.), *A Guided Tour of Artificial*
619 *Intelligence Research*, Vol. 1, Springer Verlag, 2020, Ch. 4, pp. 119–150.
- 620 [29] P. Smets, The combination of evidence in the transferable belief model, *IEEE Transactions on Pattern*
621 *Analysis and Machine Intelligence* 12 (5) (1990) 447–458.
- 622 [30] T. Denœux, A k-nearest neighbor classification rule based on Dempster-Shafer theory, *IEEE Transac-*
623 *tions on Systems, Man, and Cybernetics* 25 (5) (1995) 804–813. doi:10.1109/21.376493.
- 624 [31] T. Denœux, A neural network classifier based on Dempster-Shafer theory, *IEEE Transactions on Sys-*
625 *tems, Man, and Cybernetics-Part A: Systems and Humans* 30 (2) (2000) 131–150.
- 626 [32] T. Denœux, M.-H. Masson, Evclus: evidential clustering of proximity data, *IEEE Transactions on*
627 *Systems, Man, and Cybernetics, Part B (Cybernetics)* 34 (1) (2004) 95–109.
- 628 [33] F. Pichon, D. Mercier, E. Lefèvre, F. Delmotte, Proposition and learning of some belief function
629 contextual correction mechanisms, *International Journal of Approximate Reasoning* 72 (2016) 4–42.
- 630 [34] F. Pichon, D. Dubois, T. Denœux, Quality of information sources in information fusion, in: É. Bossé,
631 G. L. Rogova (Eds.), *Information Quality in Information Fusion and Decision Making*, Springer Inter-
632 *national Publishing, Cham*, 2019, pp. 31–49.
- 633 [35] H. Chen, S. Le Hégarat-Masclé, E. Aldea, Belief functions clustering for epipole localization, *Internat-*
634 *ional Journal of Approximate Reasoning* 137 (2021) 146–165.
- 635 [36] T. Denœux, O. Kanjanatarakul, S. Sriboonchitta, A new evidential k-nearest neighbor rule based
636 on contextual discounting with partially supervised learning, *International Journal of Approximate*
637 *Reasoning* 113 (2019) 287–302.
- 638 [37] C. Gong, Z. gang Su, P. hong Wang, Q. Wang, Y. You, Evidential instance selection for k-nearest
639 neighbor classification of big data, *International Journal of Approximate Reasoning* 138 (2021) 123–
640 144.
- 641 [38] A. Imoussaten, L. Jacquin, Cautious classification based on belief functions theory and imprecise rela-
642 belling, *International Journal of Approximate Reasoning* 142 (2022) 130–146.
- 643 [39] T. Denœux, NN-EVCLUS: Neural network-based evidential clustering, *Information Sciences* 572 (2021)
644 297–330.
- 645 [40] V. Antoine, J. A. Guerrero, J. Xie, Fast semi-supervised evidential clustering, *International Journal of*
646 *Approximate Reasoning* 133 (2021) 116–132.
- 647 [41] C. Lian, S. Ruan, T. Denœux, H. Li, P. Vera, Joint tumor segmentation in PET-CT images using co-
648 clustering and fusion based on belief functions, *IEEE Transactions on Image Processing* 28 (2) (2018)
649 755–766.
- 650 [42] L. Huang, S. Ruan, T. Denœux, Belief function-based semi-supervised learning for brain tumor seg-
651 mentation, arXiv preprint arXiv:2102.00097.
- 652 [43] Z. Tong, P. Xu, T. Denœux, Evidential fully convolutional network for semantic segmentation, *Applied*
653 *Intelligence* 51 (2021) 6376–6399.
- 654 [44] L. Huang, S. Ruan, P. Decazes, T. Denœux, Evidential segmentation of 3D PET/CT images, in:
655 T. Denœux, E. Lefèvre, Z. Liu, F. Pichon (Eds.), *Belief Functions: Theory and Applications*, Springer
656 *International Publishing, Cham*, 2021, pp. 159–167.
- 657 [45] P. Smets, R. Kennes, The Transferable Belief Model, *Artificial Intelligence* 66 (1994) 191–243.
- 658 [46] T. Denœux, Analysis of evidence-theoretic decision rules for pattern classification, *Pattern Recognition*
659 30 (7) (1997) 1095–1107.
- 660 [47] L. Ma, T. Denœux, Partial classification in the belief function framework, *Knowledge-Based Systems*
661 214 (2021) 106742.
662 URL <http://www.sciencedirect.com/science/article/pii/S0950705121000058>
- 663 [48] T. Denœux, Decision-making with belief functions: A review, *International Journal of Approximate*
664 *Reasoning* 109 (2019) 87–110.
- 665 [49] T. Denœux, Logistic regression, neural networks and Dempster-Shafer theory: A new perspective,
666 *Knowledge-Based Systems* 176 (2019) 54–67.
- 667 [50] Z. Tong, P. Xu, T. Denœux, An evidential classifier based on Dempster-Shafer theory and deep learning,
668 *Neurocomputing* 450 (2021) 275–293.

- 669 [51] E. Kerfoot, J. Clough, I. Oksuz, J. Lee, A. P. King, J. A. Schnabel, Left-ventricle quantification using
670 residual u-net, in: International Workshop on Statistical Atlases and Computational Models of the
671 Heart, Springer, 2018, pp. 371–380.
- 672 [52] D. Ulyanov, A. Vedaldi, V. Lempitsky, Improved texture networks: Maximizing quality and diversity
673 in feed-forward stylization and texture synthesis, in: Proceedings of the IEEE conference on computer
674 vision and pattern recognition, 2017, pp. 6924–6932.
- 675 [53] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on
676 imagenet classification, in: Proceedings of the IEEE international conference on computer vision, 2015,
677 pp. 1026–1034.
- 678 [54] B. Lowekamp, D. Chen, L. Ibáñez, D. Blezek, The design of SimpleITK, *Frontiers in Neuroinformatics*
679 7 (2013) 45. doi:10.3389/fninf.2013.00045.
680 URL <https://www.frontiersin.org/article/10.3389/fninf.2013.00045>
- 681 [55] Z. Yaniv, B. C. Lowekamp, H. J. Johnson, R. Beare, SimpleITK image-analysis notebooks: a collabora-
682 tive environment for education and reproducible research, *Journal of Digital Imaging* 31 (3) (2018)
683 290–303. doi:10.1007/s10278-017-0037-8.
684 URL <https://doi.org/10.1007/s10278-017-0037-8>
- 685 [56] C. Guo, G. Pleiss, Y. Sun, K. Q. Weinberger, On calibration of modern neural networks, in: Interna-
686 tional Conference on Machine Learning, PMLR, 2017, pp. 1321–1330.
- 687 [57] A. Jungo, F. Balsiger, M. Reyes, Analyzing the quality and challenges of uncertainty estimations for
688 brain tumor segmentation, *Frontiers in Neuroscience* 14 (2020) 282.
- 689 [58] A.-J. Rousseau, T. Becker, J. Bertels, M. B. Blaschko, D. Valkenborg, Post training uncertainty calibra-
690 tion of deep networks for medical image segmentation, in: 2021 IEEE 18th International Symposium
691 on Biomedical Imaging (ISBI), IEEE, 2021, pp. 1052–1056.
- 692 [59] W. J. Conover, R. L. Iman, On multiple-comparisons procedures, Tech. Rep. LA-7677-MS, Los Alamos
693 Scientific Laboratory (1979).
- 694 [60] A. Dinno, conover.test: Conover-Iman Test of Multiple Comparisons Using Rank Sums, *r* package
695 version 1.1.5 (2017).
696 URL <https://CRAN.R-project.org/package=conover.test>
- 697 [61] Y. Benjamini, D. Yekutieli, The control of the false discovery rate in multiple testing under dependency,
698 *The Annals of Statistics* 29 (4) (2001) 1165–1188.
- 699 [62] A. Hryniewski, A. Wong, Deeplabnet: End-to-end learning of deep radial basis networks, *Journal of*
700 *Computational Vision and Imaging Systems* 5 (1) (2020) 1.
701 URL <https://openjournals.uwaterloo.ca/index.php/vsl/article/view/1663>