Pós-Graduação em Ciência da Computação

Michel Mozinho dos Santos

Classificação de Padrões de Imagens:
Função Objetivo para Perceptron Multicamada e Máquina de Aprendizado Extremo Convolucional

Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
http://cin.ufpe.br/~posgraduacao

Recife
2019

Michel Mozinho dos Santos

Classificação de Padrões de Imagens:
Função Objetivo para Perceptron Multicamada e Máquina de Aprendizado Extremo Convolucional

Tese de Doutorado apresentada ao Programa de Pós-graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, como requisito parcial para obtenção do título de Doutor em Ciência da Computação.

Área de Concentração: Inteligência Computacional
Orientador: Prof. Dr. Abel Guilhermino da Silva Filho
Coorientador: Prof. Dr. Wellington Pinheiro dos Santos

Recife
2019

Catalogação na fonte
Bibliotecária Arabelly Ascoli CRB4-2068

S237c Santos, Michel Mozinho dos
Classificação de padrões de imagens: função objetivo para perceptron multicamada e máquina de aprendizado extremo convolucional / Michel Mozinho dos Santos. – 2019.
132 f.: fig., tab.

Orientador: Abel Guilhermino da Silva Filho
Tese (Doutorado) – Universidade Federal de Pernambuco. CIn. Ciência da Computação. Recife, 2019.
Inclui referências e apêndices.

1. Inteligência computacional. 2. Função objetiva específica 3. Máquina de aprendizado extremo convolucional. 4. Segmentação de lesão de esclerose múltipla. I. Silva Filho, Abel Guilhermino da (orientador). II. Título.

006.31 CDD (22. ed.) UFPE-CCEN 2020-24

Michel Mozinho dos Santos

"**Classificação de Padrões de Imagens:**
Função Objetivo para Perceptron Multicamada e Máquina de Aprendizado Extremo Convolucional"

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Doutor em Ciência da Computação.

Aprovado em: 28/08/2019.

Orientador: Prof. Dr. Abel Guilhermino da Silva Filho

BANCA EXAMINADORA

Prof. Dr. Paulo Salgado Gomes de Mattos Neto
Centro de Informática / UFPE

Prof. Dr. Francisco Marcos de Assis
Departamento de Engenharia Elétrica / UFCG

Prof. Dr. Fernando José Ribeiro Sales
Departamento de Engenharia Biomédica / UFPE

Prof. Dr. Filipe Rolim Cordeiro
Departamento de Computação / UFRPE

Prof. Dr. Sidney Marlon Lopes de Lima
Departamento de Eletrônica e Sistemas / UFPE

Dedico esta tese a minha esposa e ao nosso filho, com todo meu amor.

AGRADECIMENTOS

Gostaria de agradecer aos cidadãos brasileiros. O imposto pago pela sociedade é a contribuição que antecede e fomenta esta Tese. Que fique claro: não tive bolsa de apoio nesta etapa da minha formação. Mas, o caminho que me trouxe até aqui foi de tantas formas suportado pela sociedade que só posso dizer muito, muito obrigado!

Esta Tese não teria acontecido sem o apoio, paciência e confiança do Prof. Abel que me orientou por esta jornada. O mesmo vale para o coorientador, Prof. Wellington, que me recebeu, ainda no mestrado, em um momento difícil da minha carreira acadêmica, sendo sempre motivador e inspirador.

Professores, funcionários e alunos do Centro de Informática da UFPE, eu sou muito grato a todos vocês. A coesão e a sinergia de vocês é o que faz do CIn um centro de excelência. Além disso, reconheço o profissionalismo, a seriedade e responsabilidade que predomina nesse centro. Tive oportunidade de trilhar o doutorado no CIn, usufruindo dos serviços e da estrutura, por isso sou grato.

Também à equipe do Nutes-UFPE, à Profa. Magdala, ao Prof. Fernando, à Nilma, à Luciana e aos demais colegas, eu ofereço meu apreço e gratidão. Uma parte do trabalho de pesquisa realizado para esta Tese foi feito dentro do Nutes.

Não poderia me esquecer da pessoa mais importante. Paula, querida esposa, eu espero poder retribuir a cada dia seu apoio. Agradeço a você pelas conversas inspiradoras, pelas explicações, pelas discussões e direcionamentos sobre neurologia e, em particular, sobre a segmentação de lesões de esclerose múltipla.

Por fim, agradeço aos demais familiares. Aos meus pais, Raimundo (*in memoriam*) e Osmarina, pelo amor, incentivo, dedicação e compreensão, sem os quais eu não teria conseguido concluir este doutorado. Ao meu filho Miguel que é em si mesmo uma fonte de motivação. A minha sogra, Eliane, que em um momento crucial, nos ajudou, possibilitando que eu tivesse mais tempo para finalizar a Tese. Novamente, eu agradeço a todos que direta ou indiretamente contribuíram para este trabalho.

RESUMO

Avanços recentes em redes neurais profundas possibilitaram resolver certas tarefas de classificação de padrões de imagens com acurácia equivalente ou superior à humana. No entanto, as redes neurais profundas padecem do longo tempo de treinamento (de horas, à dias). Isto pode ser problemático em determinadas aplicações, por exemplo, se o treinamento ocorrer em uma nuvem computacional com cobrança pelo tempo de uso. Nesta tese, abordamos o problema de classificação de padrões de imagens com redes neurais artificiais em duas tarefas diferentes. Na primeira abordagem, propomos funções objetivo específicas para a tarefa de segmentação de lesões de esclerose múltipla (EM), usando uma simples rede perceptron multicamada. Mantendo o tempo do treinamento rápido, uma função objetivo específica resultou em desempenho de segmentação de EM que foi melhor do que com uma função objetivo inespecífica. No entanto, mesmo com função objetivo específica, nosso método de segmentação de EM teve desempenho de segmentação insatisfatório em relação ao estado-da-arte, devido a simplicidade do perceptron multicamada empregado. Em uma segunda abordagem, investigamos máquinas de aprendizado extremo convolucional (*convolutional extreme leaning machines*, CELM), uma alternativa para desenvolver uma rede neural convolucional de treinamento rápido e com elevada acurácia, na tarefa de reconhecimento de dígitos. Propomos uma CELM profunda com dois estágios de extração de características e duas camadas no estágio de classificação. Comparamos bancos de filtros puros e combinados, em dois estágios de extração de características. Para a CELM proposta, mostramos que o modelo de erro baseado no teorema de Rahimi-Retch é capaz de ajustar o erro empírico. Contribuímos assim para um melhor entendimento do erro de generalização em função do tamanho do conjunto de treinamento e do número de parâmetros ajustáveis, conjecturados previamente como fatores de sucesso do aprendizado profundo. Além disso, no reconhecimento de dígitos, a CELM proposta resultou em acurácia superior (99,775%), bem como em tempo de treinamento competitivo (9 min., em CPU), mesmo em relação a abordagens que empregam processamento em GPU, no conjunto de dados de reconhecimento de dígitos EMNIST.

Palavras-chaves: Função Objetivo Específica. Máquina de Aprendizado Extremo Convolucional. Segmentação de Lesão de Esclerose Múltipla. Reconhecimento de Dígito. Rede Neural Artificial.

ABSTRACT

Recent advances in deep neural networks have made possible solving image pattern classification tasks with accuracy equivalent to or higher than humans. However, deep neural networks suffer from long training time (from hours to days). The time spent can be problematic in applications, for instance, if the training takes place in a computational cloud with time-to-use charging. In this thesis, we address the problem of image pattern classification with artificial neural networks in two different tasks. In the first approach, we propose specific objective functions for the segmentation of multiple sclerosis (MS) lesions using a simple multilayer perceptron. Keeping the fast training time, a specific objective function resulted in segmentation performance that was better than with a nonspecific objective function. However, even with a specific objective function, our MS segmentation method resulted in unsatisfactory segmentation performance in relation to state-of-the-art, due to the simplicity of the multilayer perceptron employed. In the second approach, we have investigated convolutional extreme learning machines (CELMs) for digit recognition task, as an alternative for developing a CNN with fast training time. We propose a deep CELM with two stages of feature extraction and two layers in the classification stage. We compare pure and combined filter banks in two stages of feature extraction. For the proposed CELM, we show that the error model based on the Rahimi-Retch theorem can adjust the empirical error. In this way, we contribute to a better understanding of the generalization error as a function of the training set size and the number of adjustable parameters, which are two conjectured factors for the exceptional generalization capability in deep learning. Moreover, the proposed CELM for digit recognition resulted in superior accuracy (99.775 %) as well as competitive training time (9 min, in CPU), even concerning approaches using GPU processing, over the EMNIST dataset for digit recognition.

Keywords: Specific Objective Function. Convolutional Extreme Learning Machine. Segmentation of Multiple Sclerosis Lesion. Digit Recognition. Artificial Neural Network.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ASD	Average Surface Distance
ASSD	Average Symmetric Surface Distance
CELM	Convolutional Extreme Learning Machine
CKELM	Convolutional Extreme Learning Machine with Kernel
CNN	Convolutional Neural Network
DSC	Dice Similarity Coefficient
ELM	Extreme Learning Machine
ELM-LRF	Extreme Learning Machime Local Receptive Field
EM	Esclerose Múltipla
EMNIST	Extended MNIST
FLAIR	Fluid Attenuated Inversion Recovery Image
FPR	False Negative Rate
LFNR	Lesion-wise False Negative Rate
LPPV	Lesion-wise Positive Predictive Value
LTPR	Lesion-wise True Positive Rate
MICCAI2008	International Conference on Medical Image Computing and Computer Assisted Intervention of 2008
MICCAI2016	International Conference on Medical Image Computing and Computer Assisted Intervention of 2016
MLP	Multilayer Perceptron
MLP-PSO	Multilayer Perceptron trained by Particle Swarm Optimization
MNIST	Modified National Institute of Standards and Technology Database
MS	Multiple Sclerosis
MSE	Mean Squared Error
MSSEG2016	Multiple Sclerosis Lesions Segmentation Challenge 2016
PCA	Principal Component Analysis
PD	Proton Density Weighted Image
PPV	Positive Predictive Value

PSO	Particle Swarm Optimization
RAVD	Relative Absolute Volume Difference
RKS	Random Kitchen Sinks
RM	Ressonância Magnética
Se	Sensitivity
Sp	Specificity
T1	T1 weighted Image
T2	T2 weighted Image
TLL	Total Lesion Load
TNR	True Negative Rate
TPR	True Positive Rate

SUMÁRIO

1 INTRODUÇÃO

Antes de iniciar os assuntos da presente tese, definimos nosso leitor alvo. Assim, este texto supõe que o leitor possui conhecimentos compatíveis com a formação em um curso superior em Ciência da Computação ou áreas afins. Após esta consideração, começamos apresentando o contexto e algumas definições básicas, na seção logo abaixo.

1.1 CONTEXTO E DEFINIÇÕES BÁSICAS

O ser humano tem a habilidade de categorizar objetos do mundo real a partir de características. A visão tem um papel proeminente no processo de categorização. Cor, forma e brilho são alguns exemplos de atributos visuais que são naturalmente empregados pelos seres humanos para distinguir objetos. Com o desenvolvimento tecnológico, sistemas computacionais que emulam a capacidade humana de classificar objetos em imagens são de extremo interesse na sociedade contemporânea (GONZALEZ; WOODS, 2018).

Em Aprendizado de Máquina, um *algoritmo de classificação* (ou classificador) serve para identificar a qual classe pertence um objeto. A classificação supervisionada envolve um *conjunto de dados* de exemplos, uma *função de mapeamento* (ou *modelo ajustável*), uma *função objetivo* (ou *função de custo*) e um *algoritmo de treinamento* (GOODFELLOW; COURVILLE; BENGIO, 2016, chapter 5). Compondo o conjunto de dados, cada exemplo é uma representação numérica que consiste em um vetor de característica do objeto, além de um rótulo indicando a classe. A função de mapeamento possui parâmetros que são ajustados, a fim de encontrar um mapeamento das características (entradas) para a classe predita (saída). Em geral, o número de parâmetros reflete a capacidade ou o poder expressivo do modelo. Dado um determinado conjunto de parâmetros, o sucesso do mapeamento é quantificado através da função objetivo. Deste modo, o algoritmo de treinamento ajusta os parâmetros da função de mapeamento otimizando a função objetivo. Após o treinamento, a função de mapeamento é idealmente capaz de categorizar novos objetos a partir das características extraídas. Além disso, dois conceitos são importantes para descrever o desempenho de um algoritmo de classificação. O primeiro é o conceito de *generalização* que acontece quando o erro em um conjunto de dados inédito é tão baixo quanto o erro obtido sob o conjunto treinamento. Por outro lado, o *overfitting* (ou sobreajuste) ocorre quando o modelo memoriza o conjunto de treinamento, obtendo alto erro em um conjunto de dados novos.

Redes neurais artificiais (RNA) constituem uma classe de algoritmos de aprendizado de máquina com inspiração biológica (BISHOP, 1996; HAYKIN, 2008). Uma RNA é composta por unidades de processamento denominadas neurônios artificiais. Matematicamente, um neurônio é representado por uma função de ativação que recebe um conjunto de valores

de entrada. As entradas são ponderas por parâmetros livres que representam os pesos ou conexões sinápticas. A saída do neurônio é um valor que representa o grau de ativação associado ao estímulo de entrada. Os neurônios são dispostos em camadas, em que cada camada recebe na entrada os sinais da camada anterior. O vetor de entrada é frequentemente tratado como uma camada. Enquanto, a última camada representa a saída desejada, muitas vezes na forma de probabilidades associadas a cada classe. As camadas responsáveis efetivamente pelo processamento ficam entre as camadas de entrada e de saída, sendo por isso denominadas camadas ocultas.

A rede *perceptron multicamada* (*multilayer perceptron*, MLP) possui uma ou mais camadas ocultas com fluxo de processamento ocorrendo em uma única direção, isto é, sem ciclos até a saída, sendo por isto também chamada de rede de alimentação para a frente (*feed-forward network*) (BISHOP, 2006). Além disso, as camadas são totalmente conectadas (*fully connected*), isto é, cada neurônio recebe conexões de todos os neurônios da camada anterior (Figura 1).

O treinamento de um perceptron multicamada é tipicamente realizado através do algoritmo de retropropagação (*backpropagation*), baseado no método de descida de gradiente. Uma vez treinado, o perceptron multicamada é relativamente rápido para computar a saída, sendo vantajoso em relação a outros métodos de aprendizado de máquina. Como desvantagem na classificação de padrões de imagens, o uso de camadas completamente conectadas requer um grande número de pesos por neurônio.

A máquina de aprendizado extremo (*extreme learning machine*, ELM) é um algoritmo de treinamento rápido para redes de alimentação para frente com uma única camada oculta (HUANG; ZHU; SIEW, 2006). Os pesos dos neurônios ocultos são gerados aleatoriamente, enquanto os pesos na camada de saída são calculados analiticamente pelo método de mínimos quadrados. Apresentando bom desempenho de generalização, este algoritmo de aprendizado é até milhares de vezes mais rápido que abordagens tradicionais, como retropropagação (*backpropagation*) (RUMELHART; HINTON; WILLIAMS, 1986) e máquina de vetor de suporte (*support vector machine*) (VAPNIK, 2000).

A otimização por exame de partículas (*particle swarm optimization* , PSO) é um método estocástico de busca que pode ser usado para treinamento de perceptrons multicamada (KENNEDY; EBERHART, 1995; EBERHART; SHI, 2007). O PSO permite maior liberdade na escolha da função objetivo para uma dada tarefa de aprendizado. Em contraste, algoritmos de treinamento como retropropagação, máquina de vetor de suporte e máquina de aprendizado extremo, restringem a forma da função objetivo, por exemplo, ao erro quadrático médio, o que pode não ser apropriado para uma dada tarefa de aprendizado.

A *rede neural convolucional* (*convolutional neural networks*, CNN) é disposta em múltiplas camadas alimentadas para a frente, podendo ser vista como uma classe especial de perceptron multicamada (HAYKIN, 2008; LECUN et al., 1998). Diferente do perceptron

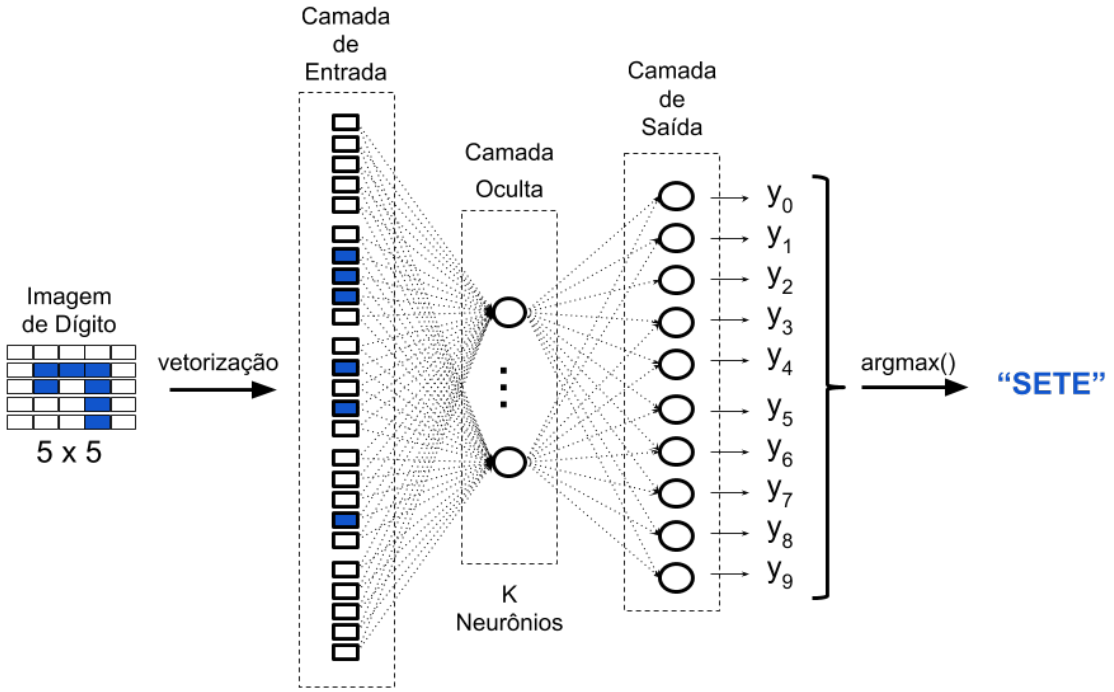

Figura 1 – Arquitetura do perceptron multicamada para reconhecimento de dígitos em imagens de tamanho 5×5. A rede neural recebe a imagem vetorizada do dígito sete na camada de entrada. A camada oculta contém K neurônios completamente conectados, com apenas o primeiro e o último neurônios sendo mostrados, por simplicidade. Na camada de saída, cada neurônio é também completamente conectado e representa um dígito possível. O neurônio de saída cujo valor de ativação é máximo, obtido pela função *argmax()*, corresponde a classe apontada pela rede neural, nesse caso predizendo corretamente a classe "SETE". Note que como o fluxo de processamento é acíclico, isto é, ocorrendo em um único sentido, essa rede neural é chamada de alimentada para a frente.

multicamada típico, a estrutura de uma rede neural convolucional é projetada para reconhecer objetos em imagens, possuindo camadas bidimensionais de neurônios. Ao invés de neurônios completamente conectados, em uma camada convolucional, cada neurônio recebe entradas somente de uma região restrita da imagem chamada de campo receptivo local (*local receptive field*) para realizar extração de características (*feature extraction*). Outra diferença é o compartilhamento de pesos entre os diferentes campos receptivos locais, o que reduz o número de pesos a serem ajustados. Deste modo, um único conjunto de pesos, isto é, um filtro convolucional, é aplicado em diferentes campos receptivos locais ao longo da imagem de entrada, dando origem a uma nova imagem chamada de mapa de característica (*feature map*). Uma camada convolucional pode resultar em múltiplos

mapas de características, um para cada filtro convolucional. Além disso, uma camada de subamostragem (*pooling*) computa uma média local pondera por pesos ajustáveis para reduzir o tamanho dos mapas de características. A Figura 2 ilustra uma rede neural convolucional para reconhecimento de dígitos. Este tipo de rede neural tem sido usado com sucesso para resolver problemas de classificação de padrões de imagens considerados difíceis (LECUN; BENGIO; HINTON, 2015). O aprendizado profundo (*deep leaning*) é um

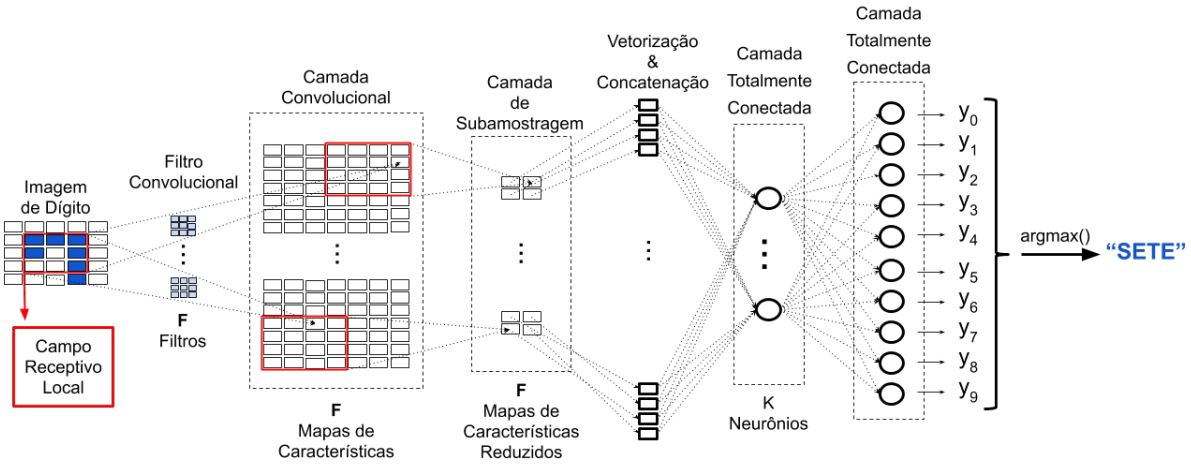

Figura 2 – Arquitetura da rede neural convolucional para reconhecimento de dígitos em imagens de tamanho 5×5. Na camada convolucional são representados F mapas de características resultando da convolução da imagem de entrada com cada um de F filtros convolucionais. Na camada de subamostragem, estão os mapas de características reduzidos após a operação de subamostragem, que consiste na convolução de cada mapa de características com um único filtro de subamostragem (não mostrado). Os mapas de características são vetorizados e concatenados servindo como entradas da primeira camada totalmente conectada (da esquerda para a direita). Então, a segunda camada totalmente conectada recebe os dados da camada anterior e gera saídas para cada classe de dígitos. Por fim, a classe de saída é apontada através da função *argmax()*

campo na área de RNAs que lida com o treinamento de modelos com um grande número de camadas. Além disso, a arquitetura de uma rede de aprendizado profundo pode mesclar tipos diferentes de camadas como convolucional, de subamostragem e completamente conectada. Atualmente, há muito interesse em aprendizado profundo, despertado por trabalhos com enfoque prático que foram capazes de superar o desafio do treinamento de modelos com várias camadas, alcançando níveis de desempenho sem precedentes, em tarefas de classificação de padrões de imagens (KRIZHEVSKY; SUTSKEVER; HINTON, 2012; CIREŞAN; MEIER; SCHMIDHUBER, 2012).

Os fatores que tornam o aprendizado profundo bem-sucedido ainda não estão claros (ZHANG et al., 2016). A visão tradicionalmente aceita atribui um pequeno erro de genera-

lização ao controle do número de parâmetros do modelo, através da limitação da própria capacidade ou do uso de técnicas de regularização (ZHANG et al., 2016). Mas essas abordagens são contrárias ao fato de que no aprendizado profundo os modelos do estado-da-arte resultam em um grande número de parâmetros. Em uma visão alternativa, no trabalho de Krizhevsky, Sutskever e Hinton (2012), os autores provêm justificativas plausíveis para o sucesso de redes neurais profundas:

1. **grandes conjuntos de dados** capazes de refletir a variabilidade presente em tarefas como o reconhecimento de objetos em imagens naturais;

2. **grande capacidade de representação**, isto é, modelos com um vasto número de parâmetros ajustáveis, e além disso, fazendo suposições razoáveis sobre a natureza das imagens como a localidade nas dependências entre elementos da imagem (*pixels*).

3. **poder de processamento suficiente** para tratar tarefas de larga escala e com imagens de alta resolução, o que tornou-se viável por meio do uso de unidades de processamento gráfico (*graphic processing units*, GPUs).

Na classificação de padrões de imagens, as tarefas de reconhecimento e segmentação podem ser resolvidas com modelos de RNAs similares que diferem no uso da informação espacial na saída da rede (LONG; SHELHAMER; DARRELL, 2015; GONZALEZ; WOODS, 2018). Por um lado, a tarefa de reconhecimento visa identificar qual objeto está presente na imagem, sem a necessidade de determinar a posição exata do objeto. Assim, o reconhecimento é realizado atribuindo-se para cada imagem como um todo uma única classe. Por outro lado, na tarefa de segmentação, a posição do objeto na imagem deve ser determinada. Para isto, a segmentação atribui uma classe para cada posição ou elemento da imagem. O resultado esperado é a demarcação de regiões contíguas da imagem. A Figura 3 ilustra as tarefas de reconhecimento e segmentação realizadas com dois classificadores similares baseados no mesmo tipo de rede neural convolucional (LONG; SHELHAMER; DARRELL, 2015). Assim, embora as duas tarefas sejam diferentes, um avanço em um classificador neural para reconhecimento pode ser útil também em um classificador neural análogo para segmentação e vice-versa. A seguir, apresentamos uma aplicação específica de cada uma dessas duas tarefas, destacando o estado de desenvolvimento atual.

Por um lado, apesar de bastante investigada, a segmentação de lesões de esclerose múltipla poder ser considerada um problema em aberto (GARCÍA-LORENZO et al., 2013). Como argumento neste sentido, na competição do MICCAI2008 (STYNER et al., 2008), a abordagem atualmente melhor classificada (VALVERDE et al., 2017) tem desempenho de segmentação de 87,11%, abaixo do esperado para um especialista humano que é 90%. Em outra competição, o MICCAI2016 (COMMOWICK et al., 2018), nenhum algoritmo atingiu o desempenho de segmentação de um especialista humano.

Por outro lado, o reconhecimento de dígitos manuscritos foi um dos primeiros casos de um problema do mundo real solucionado com sucesso por uma rede neural convolucional

(LECUN et al., 1990). Inúmeros algoritmos de classificação foram aplicados ao conjunto de dados de reconhecimento de dígitos MNIST, servindo como referência amplamente aceita. Atualmente, o MNIST não é considerado desafiador, com vários pesquisadores tendo reportado acurácias acima de 99%, por exemplo (WAN et al., 2013; CIREŞAN; MEIER; SCHMIDHUBER, 2012; MCDONNELL; VLADUSICH, 2015). Recentemente, o conjunto de dados EMNIST foi elaborado para fornecer uma tarefa de reconhecimento mais difícil (COHEN et al., 2017).

Nesta tese abordamos o problema de classificação de padrões de imagens com redes neurais artificiais, tratando duas tarefas diferentes. Primeiramente, propomos uma *função objetivo* para a tarefa de segmentação de lesões de esclerose múltipla, empregando um simples modelo perceptron multicamada. Além disso, propomos uma *máquina de aprendizado extremo convolucional profunda* para a tarefa de reconhecimento de dígitos, avaliamos filtros convolucionais predefinidos e combinações, bem como validamos um modelo de erro de generalização.

O restante deste capítulo está organizado como descrito a seguir. A Seção 1.2 introduz as nossas principais motivações através de aplicações. Depois, a Seção 1.3 declara o problema de pesquisa que será abordado nesta tese. Então, na Seção 1.4, são apresentadas as razões para a escolha das redes neurais e algoritmos de treinamento investigados neste trabalho.

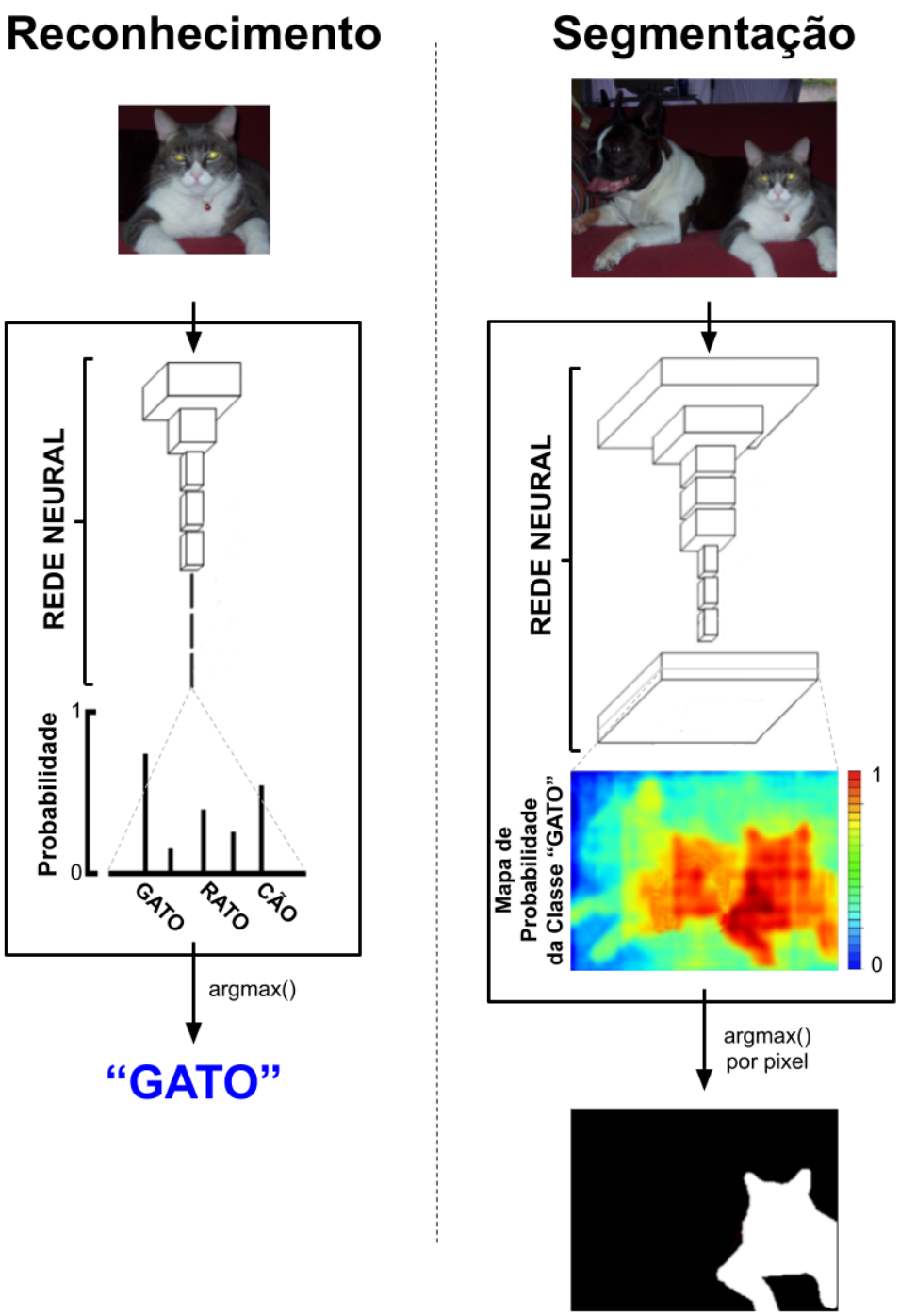

Figura 3 – Classificadores neurais similares aplicados em duas tarefas distintas: reconhecimento e segmentação. À esquerda, no reconhecimento, a classificação trata a imagem como um todo, desconsiderando a informação espacial na classe atribuída. À direita, na segmentação, a classificação acontece para cada ponto da imagem, determinando uma classe para cada posição espacial (adaptado de (LONG; SHELHAMER; DARRELL, 2015; HUANG et al., 2017b)).

1.2 MOTIVAÇÃO

A esclerose múltipla é uma doença neurodegenerativa (COMPSTON; COLES, 2008), incapacitante em 60% dos casos. Figurando como uma das doença neurológicas mais frequentes, a EM afeta cerca de 2,3 milhões de pessoas no mundo (VOS et al., 2015). A EM é caracterizada por regiões com desmielinização que são detectáveis em imagens de ressonância magnética (RM) (HAUSER; GOODKIN, 2001; POLMAN et al., 2011). A quantificação destas lesões têm importância fundamental como ferramenta para o diagnóstico e a progressão da EM (BARKHOF et al., 1997). Os métodos baseado em redes neurais convolucionais em (VALVERDE et al., 2017; MCKINLEY et al., 2016) representam o estado-da-arte na segmentação de lesões de EM.

O reconhecimento de dígitos é uma tarefa mais simples que a segmentação de lesões de EM para o estudo de classificadores neurais. As principais diferenças entre essas duas tarefas estão relacionadas ao tamanho das imagens, ao número de exemplos e ao pré-processamento. Quanto ao tamanho das imagens, enquanto a segmentação de lesões de EM pode envolver de 1 a 5 imagens de tamanho $512 \times 512 \times 512$ por paciente, já o reconhecimento de dígitos manuscritos envolve 1 única imagem 28×28 por dígito, o que é aproximadamente até seis ordens de grandeza menor no número de elementos. Quanto ao número de exemplos, por um lado, os conjuntos de dados de segmentação de EM contém cerca de 20 pacientes (STYNER et al., 2008; COMMOWICK et al., 2018), o que pode representar pouco a variabilidade entre indivíduos. Por outro lado, o conjunto de dado de reconhecimento de dígitos contém cerca de 200.000 dígitos (COHEN et al., 2017). Note que o tamanho de um conjunto de dados não-compactado para treinamento do classificador é da ordem de 20 *gigabytes* na segmentação de lesões de EM e, em contraste, da ordem de 0,2 *gigabytes* no reconhecimento de dígitos. Quanto ao pré-processamento, problemas de aquisição comuns em imagens de RM, como o artefato de intensidade não-uniforme, requerem etapas de pré-processamento para segmentação de lesões de EM que não são necessárias para o reconhecimento de dígitos.

1.3 PROBLEMA DE PESQUISA

Idealmente, um classificador neural deve ser treinado rapidamente e ter nível de desempenho equiparável ou superior ao humano em uma dada tarefa. Classificadores neurais profundos representam o estado-da-arte em reconhecimento de dígitos (CIREŞAN; MEIER; SCHMIDHUBER, 2012; WAN et al., 2013; SATO; NISHIMURA; YOKOI, 2015) e em segmentação de lesões de EM (VALVERDE et al., 2017; MCKINLEY et al., 2016; COMMOWICK et al., 2018), contudo atingindo apenas na primeira tarefa desempenho próximo ao humano que tem erro de 0,2% (LECUN et al., 1995).

Apesar do sucesso, as redes neurais profundas levam um longo tempo de treinamento (de horas a dias), o que pode ser um obstáculo em situações que requerem treinamento

recorrente ou em tempo (quase) real, como no caso da avaliação estatística do método de classificação ou no caso de treinamento na nuvem com cobrança por tempo de alocação de recursos computacionais. Assim, podemos declarar o problema de pesquisa: **de que maneira um método de classificação de padrões de imagens, por redes neurais artificiais, pode apresentar treinamento rápido e, ao mesmo tempo, atingir ou superar o nível de desempenho humano em uma dada tarefa de classificação?**

1.4 JUSTIFICATIVAS INICIAIS

A seguir apresentamos justificativas para a escolha das redes neurais e algoritmos de treinamento abordados neste trabalho.

A rede MLP foi aplicada anteriormente à segmentação de lesões de EM com relativo sucesso (ZIJDENBOS; FORGHANI; EVANS, 2002). Métricas de segmentação como função objetivo podem melhorar o desempenho de segmentação (LECOEUR; FERRÉ; BARILLOT, 2009). Em geral, redes neurais para segmentação de lesões de EM são otimizadas quanto ao erro quadrático, mas não quanto as métricas de segmentação usadas na avaliação dos resultados de segmentação (ZIJDENBOS; FORGHANI; EVANS, 2002; VAIDYA et al., 2015).

Em alternativa ao treinamento por *retro-propagação*, o PSO pode ser usado para treinar rapidamente um MLP, provendo maior liberdade na escolha da função objetivo (KENNEDY; EBERHART, 1995; EBERHART; SHI, 2007). Em contraste, os algoritmos de treinamento baseados em descida de gradiente são específicos para cada função objetivo fixada. Assim, se um algoritmo baseado em gradientes apropriado não estiver disponível, é necessário calcular os gradientes analiticamente para cada função objetivo de interesse, o que nem sempre é viável. Além disso, a otimização de uma função objetivo específica para segmentação de EM por descida de gradiente é dificultada também pela necessidade da função objetivo ser contínua e diferenciável (BISHOP, 2006), o que não parece ser diretamente o caso de funções objetivo descritas em linguagem de alto nível usando desvios condicionais e laços. Por outro lado, com o PSO, podemos otimizar funções objetivo que não precisam ser contínuas e diferenciáveis e podem ser especificadas em linguagem de alto nível. Note que outros algoritmos de aprendizado não baseados em descida de gradiente poderiam ser usados ao invés do PSO, como algoritmos genéticos, otimização por colônia de formigas e a otimização baseada em biogeografia (GAO et al., 2018). Contudo, neste trabalho, voltamos nossa atenção à especificação da função objetivo para treinamento de perceptrons multicamada e não ao algoritmo de aprendizado.

Uma desvantagem das CNNs está no longo tempo de treinamento, que pode levar até dias (CIREŞAN et al., 2011; CIREŞAN; MEIER; SCHMIDHUBER, 2012; DUFOURQ; BASSETT, 2017). Uma alternativa às CNNs treinadas através de descida de gradiente, as máquinas de aprendizado extremo convolucionais (*Convolutional Extreme Learning Machines*, CELM) são CNNs de treinamento rápido (MCDONNELL; VLADUSICH, 2015; HUANG et al., 2015b; PANG; YANG, 2016; HUANG et al., 2017a; DING; GUO; HOU, 2017; YOUSEFI-AZAR; MCDON-

NELL, 2017). A acurácia das CELMs tem se mostrado não a melhor, mas competitiva em relação ao estado-da-arte. Assim, melhorias na acurácia das CELMs são desejáveis para aplicação em tarefas de classificação de padrões, idealmente sem aumentar o tempo de treinamento.

1.5 ORGANIZAÇÃO DA TESE

O restante desta tese está organizado como descrito a seguir. O Capítulo 2 contém conceitos básicos sobre rede neural artificial, imagem de ressonância magnética e segmentação de lesões de esclerose múltipla. O Capítulo 3 analisa a literatura sobre função objetivo para segmentação de lesões de EM abrangendo redes neurais e outras abordagens, bem como revisa trabalhos anteriores sobre máquina de aprendizado extremo convolucional para reconhecimento de dígitos, abordando arquitetura, filtros convolucionais, modelo de erro de generalização e conjunto de dados de reconhecimento de dígitos EMNIST. O Capítulo 4 define questões de pesquisa, declara hipóteses e objetivos. Após, o Capitulo 5 detalha uma nova função objetivo para segmentação de lesões de EM com MLP e, também, uma nova arquitetura de rede CELM para reconhecimento de dígitos, bem como apresenta materiais e métodos empregados. O Capítulo 6 descreve experimentos e resultados envolvendo os algoritmos propostos. O Capítulo 7 encerra a tese, retomando o objetivo de pesquisa, sumarizando os resultados, discutindo limitações e implicações, apontando trabalhos futuros e listando contribuições e publicações realizadas.

2 FUNDAMENTOS

Neste capítulo, fornecemos alguns conceitos básicos sobre: rede neural artificial, na Seção 2.1; imagem de ressonância magnética, na Seção 2.2; e segmentação algorítmica de lesões de esclerose múltipla, na Seção 2.3.

2.1 REDE NEURAL ARTIFICIAL

As redes neurais artificiais constituem um paradigma de aprendizado de máquina. Nesta seção fornecemos mais detalhes sobre essa família de modelos de aprendizado.

2.1.1 Perceptron Multicamada

A rede perceptron multicamada pode ser empregada em problemas de classificação supervisionada, nos quais a camada de entrada recebe as instâncias do vetor de características $\mathbf{x}$, enquanto a camada de saída é responsável por indicar a classe. A representação dos exemplos de cada classe pode ser feita através da *codificação 1-de-C*. Formalmente, seja $\mathbf{x} \in \mathbb{R}^N$ uma instância dos dados de entrada de uma RNA, e seja $\mathbf{d} = \{d_i; i = 1, \ldots, C\}$ um vetor indicador dos rótulos atribuídos a cada classe $\mathcal{C}_i$, a codificação 1-de-C é tal que:

$$d_i = \begin{cases} 1, & \text{se } \mathbf{x} \in \mathcal{C}_i, \\ 0, & \text{caso contrário.} \end{cases} \tag{2.1}$$

A saída da RNA deve corresponder ao esquema 1-de-C. Assim, o número de neurônios na camada de saída deve ser igual ao número de classes. Além disso, uma função de ativação que produz valores no intervalo $[0, 1]$ pode ser empregada conjuntamente com um critério de discriminação para indicar a classe de saída. Um possível discriminante, é o *critério vencedor-leva-tudo* (VLT), no qual é atribuído valor 1 (um) à maior saída da rede para indicar a classe, enquanto as demais saídas recebem 0 (zero) (BISHOP, 1996).

A função *softmax* foi proposta por Bridle (1990) como modo de possibilitar a interpretação das saídas de uma RNA como probabilidades, uma vez que o uso da ativação logística por si só, em um problema de classificação de múltiplas saídas, não garante que a soma de todas as saídas seja 1 (um). Tal função preserva a ordem dos valores de entrada e pode ser considerada como um indicador suave do máximo, isto é, uma versão diferenciável de argmax(). Tal aproximação suave da função *argmax()*, para um conjunto de variáveis $\{z_i, i = 1, \ldots, C\}$, é definida como uma transformação exponencial normalizada:

$$f(z_i) = \frac{exp(\alpha z_i)}{\sum_{j=1}^{C} exp(\alpha z_j)}, \tag{2.2}$$

em que o parâmetro $\alpha > 0$ define o grau de declividade, isto é, quanto maior α mais abrupta é a variação dessa função (BISHOP, 1996).

No treinamento usual do MLP, os pesos da rede são ajustados de modo a minimizar o erro quadrático médio (*Mean Squared Error*, MSE). Sejam o_{ij} as saídas preditas pela RNA e d_{ij} as saídas esperadas, os índices $\{i = 1, \ldots, C\}$, para i-ésimo neurônio da camada de saída, e $\{j = 1, \ldots, M\}$, para o j-ésimo elemento de um conjunto de padrões de entrada sendo avaliado, o MSE pode ser definido como:

$$\frac{1}{MC} \sum_{j=1}^{M} \sum_{i=1}^{C} (o_{ij} - d_{ij})^2. \tag{2.3}$$

A minimização do MSE comumente é realizada através de um método de treinamento baseado em descida de gradiente, chamado *backpropagation* (HAYKIN, 2001).

2.1.1.1 Treinamento por Enxame de Partículas

A otimização por enxame de partículas consiste em um algoritmo meta-heurístico, que foi inspirado pelo comportamento coletivo de um bando de pássaros e tem como uma das primeiras aplicações o treinamento de redes MLP (KENNEDY; EBERHART, 1995). Em um enxame de tamanho N, cada partícula $i = \{1, \ldots, N\}$ é associada a uma posição $\mathbf{x}_i \in \mathbb{R}^M$ e a uma velocidade $\mathbf{v}_i \in \mathbb{R}^M$ referentes ao estado da partícula ao percorrer o espaço de busca com M dimensões, na iteração $k = \{1, \ldots, S\}$. A medida que as partículas percorrem o espaço de busca, a melhor posição de cada partícula e a melhor posição global são armazenadas em $\boldsymbol{p}_i \in \mathbb{R}^M$ e $\boldsymbol{g} \in \mathbb{R}^M$, respectivamente. Para isto, o valor da função objetivo em cada ponto do espaço de busca serve para comparar as partículas. Durante a execução do PSO, a atualização da velocidade de uma dada partícula i é influenciada pela melhor posição da própria partícula $\boldsymbol{p}_i$ e pela melhor posição global $\boldsymbol{g}$, na dimensão $j = \{1, \ldots, M\}$ do espaço de busca, de acordo com a equação:

$$v_{i,j}^{(k+1)} = w v_{i,j}^{(k)} + c_1 r_p (p_{i,j}^{(k)} - x_{i,j}^{(k)}) + c_2 r_g (g_j^{(k)} - x_{i,j}^{(k)}), \tag{2.4}$$

em que w (peso de inércia), c_1 e c_2 são parâmetros reais, r_p e r_g são números aleatórios uniformes entre $[0, 1]$. Além disso, as posições são atualizadas a cada iteração como segue:

$$x_{i,j}^{(k+1)} = x_{i,j}^{(k)} + v_{i,j}^{(k)}. \tag{2.5}$$

Nas primeiras aplicações do PSO, foi constatado que as velocidades rapidamente alcançavam valores muito grandes, especialmente para partículas distantes da melhor posição global e local, levando as partículas para fora do espaço de busca (ENGELBRECHT, 2007, Sec. 16.3.1). Por esta razão, usamos regras para limitar a velocidade e a posição das partículas. Quando uma partícula excede um limiar de velocidade, sua velocidade atual é redefinida para a velocidade máxima. Além disso, quando alguma partícula sai do espaço

de busca, sua posição é redefinida para a borda e sua velocidade é invertida e reduzida. Essa é a versão do PSO para otimização global encontrada em (EBERHART; SHI, 2007).

Em uma das possíveis abordagens, para aplicar o algoritmo PSO para treinar uma rede neural, cada vetor de posição de partícula contém os pesos da rede e representa um ponto no espaço de busca. Deste modo, cada posição visitada por uma partícula define uma parametrização da rede, cuja avaliação é realizada calculando-se uma medida de erro para todos os padrões de entrada do conjunto de dados em questão. Assim, a medida de erro usada para avaliar o desempenho da rede corresponde ao valor da função objetivo para a respectiva partícula. O pseudocódigo de treinamento do MLP com PSO (MLP-PSO) é apresentado no Algoritmo 1.

Algorithm 1 MLP-PSO

```
1:
2: enquanto  não condições_de_término faça
3:     para cada partícula i faça
4:         atualize velocidade v_i e posição x_i
5:         aplique restrições à velocidade v_i e posição x_i
6:         compute função objetivo com MLP
7:         atualize melhor posição global g e de cada partícula p_i
8:     fim para
9: fim enquanto
```

Comparado a algoritmos de retropropagação e outros métodos de treinamento, o PSO frequentemente é capaz de atingir menores valores de erro e mais rapidamente, isto é, com menor número de avaliações da função objetivo (GUDISE; VENAYAGAMOORTHY, 2003).

2.1.2 Máquina de Aprendizado Extremo

A máquina de aprendizado extremo (*extreme learning machine*, ELM) é um algoritmo de treinamento para redes alimentadas para frente consistindo de camadas de entrada, oculta e de saída, como na Figura 1. A principal ideia empregada no algortmo ELM é gerar aleatoriamente os parâmetros (pesos e bias) da camada oculta e obter os pesos da camada de saída de modo determinístico, por exemplo resolvendo um sistema linear através do método da matriz pseudo-inversa (HUANG; ZHU; SIEW, 2006). Dados um conjunto de entrada disposto nas matrizes $\mathbf{X}$ para características e $\mathbf{Y}$ para a saída desejada, uma função de ativação $\phi()$, a pseudo-inversa da matriz $\mathbf{A}$ denotada por $\mathbf{A}^\dagger$, a ELM pode ser representada pelo Algoritmo 2.

Algorithm 2 Máquina de Aprendizado Extremo

1: gere aleatoriamente pesos ocultos $\mathbf{W}_{\text{hidden}}$
2: compute $\mathbf{A} = \phi(\mathbf{W}_{\text{hidden}}, \mathbf{X})$
3: obtenha a matriz inversa $\mathbf{A}^\dagger$
4: calcule os pesos de saída $\mathbf{W}_{\text{out}} = \mathbf{Y}\mathbf{A}^\dagger$
5: verifique as classes aprendidas $\mathbf{Y}_{\text{predicted}} = \mathbf{W}_{\text{out}}\mathbf{A}^\dagger$
6: estime as classes para novos dados $\mathbf{Y}_{\text{new}} = \mathbf{W}_{\text{out}}\phi(\mathbf{W}_{\text{hidden}}, \mathbf{X}_{\text{new}})^\dagger$

Em relação às redes neurais rasas treinadas com métodos baseados em descida de gradiente, a ELM apresenta como vantagens: melhor capacidade de generalização e treinamento mais rápido (LIU; GAO; LI, 2012).

2.1.3 Rede Neural Convolucional

Nos últimos anos, modelos de rede neurais convolucionais profundas tonaram-se o estado-da-arte em diversas tarefas de visão computacional como: reconhecimento de números de placas, objetos e dígitos (REN et al., 2015; RAZAVIAN et al., 2014; SCHWARZ; SCHULZ; BEHNKE, 2015; LIANG; HU, 2015; SCHERER; MÜLLER; BEHNKE, 2010; AGRAWAL; GIRSHICK; MALIK, 2014; GIRSHICK et al., 2014; ZHOU et al., 2014; EITEL et al., 2015; JIA et al., 2014; BA; MNIH; KAVUKCUOGLU, 2014; LIN; ROYCHOWDHURY; MAJI, 2015; YANG; CHOI; LIN, 2016). Também começam a surgir diversas aplicações de redes convolucionais na análise de sinais e imagens médicas e no desenvolvimento de sistemas inteligentes para suporte ao diagnóstico (SHIN et al., 2016; FRITSCHER et al., 2016; SCHLEGL et al., 2015; RAVISHANKAR et al., 2016).

Uma desvantagem das CNNs está no longo tempo de treinamento, que pode levar dias (CIREŞAN et al., 2011; CIREŞAN; MEIER; SCHMIDHUBER, 2012; DUFOURQ; BASSETT, 2017), relacionado ao ajuste de pesos com métodos iterativos baseados em descida de gradiente. Isto pode ser um obstáculo em aplicações que requerem treinamento frequente ou quase em tempo real.

Uma rede neural convolucional é estruturada em múltiplos estágios de processamento que envolvem extração de características e classificação (LECUN et al., 1998; LECUN; BENGIO; HINTON, 2015). Um estágio de extração é subdividido em camada de convolução e camada de *pooling*. Em uma camada de convolução, cada imagem de entrada é convoluída com um banco de filtros, resultando em novas imagens chamada de mapas de características. Um filtro convolucional é formado por uma matriz de pesos que pode ser visto como um pequeno padrão a ser extraído da imagem de entrada. Após a convolução, cada mapa de características é submetido à uma função de ativação. Então, a camada de *pooling* realiza a subamostragem de cada mapa resultante, reduzindo dimensão e criando invariância a pequenas distorções e deslocamentos. O encadeamento de múltiplos estágios de extração permite uma representação hierárquica de características com níveis crescentes de abstração e complexidade (LECUN et al., 1990). Em uma CNN profunda, dois ou mais estágios de extração são encadeados. Após o último estágio de extração, o estágio de classificação mapeia características em rótulos de classe, por exemplo, através de uma ou mais camadas de neurônios completamente conectados. O treinamento típico de uma CNN ajusta filtros convolucionais e demais pesos por meio da retropropagação de gradientes.

2.1.4 Máquina de Aprendizado Extremo Convolucional

Máquinas de aprendizado extremo convolucionais (*Convolutional Extreme Learning Machines*, CELM) são CNNs de treinamento rápido que evitam métodos baseados em gradiente, definindo filtros de modo eficiente na extração de características e obtendo pesos por mínimos quadrados na camada de saída do estágio de classificação (MCDONNELL; VLADUSICH, 2015; HUANG et al., 2015b; PANG; YANG, 2016; HUANG et al., 2017a; DING; GUO; HOU, 2017; YOUSEFI-AZAR; MCDONNELL, 2017). Em tempo de treinamento menor (da ordem de minutos), a acurácia de classificação dessas abordagens tem se mostrado não a melhor, mas competitiva em relação ao estado-da-arte.

2.2 IMAGEM DE RESSONÂNCIA MAGNÉTICA

O imageamento por ressonância magnética (RM) é um método não-invasivo que permite a aquisição de imagens do interior do corpo humano (CONSTANTINIDES, 2016). A formação da imagem de RM é baseada em um campo magnético, bem como na emissão e recepção de sinais de radiofrequência. Na presença de campo magnético alto, o momento magnético de certos núcleos atômicos tende a ficar alinhado ao campo magnético. Então, com base na frequência de ressonância do átomo alvo, sequências ordenadas de pulsos e gradientes de radiofrequência são emitidas e causam a reorientação dos momentos nucleares em relação ao campo. Cessado o estímulo, os núcleos emitem radiofrequência ao realinhar o momento com o campo. Estes sinais de radiofrequência emitidos pelos núcleos é que são usados para formar a imagem (CONSTANTINIDES, 2016; CHAVHAN, 2013).

Conforme os parâmetros da sequência de pulsos e gradientes de radiofrequência usados para aquisição da imagem, diferentes características físicas dos tecidos podem ser capturadas, possibilitando imagens com diferentes contrastes entre tecidos (CONSTANTINIDES, 2016; CHAVHAN, 2013). Na Figura 4, são apresentadas três fatias (uma imagem de RM pode conter centenas de fatias) de três tipos de sequências de aquisição (imagens FLAIR,T1 e T2) de um único paciente. O conjunto de dados resultante de duas ou mais sequências de aquisição é designado como imagem multi-sequência. Além deste, os termos multicanal, multimodal, multi-espectral também ocorrem na literatura.

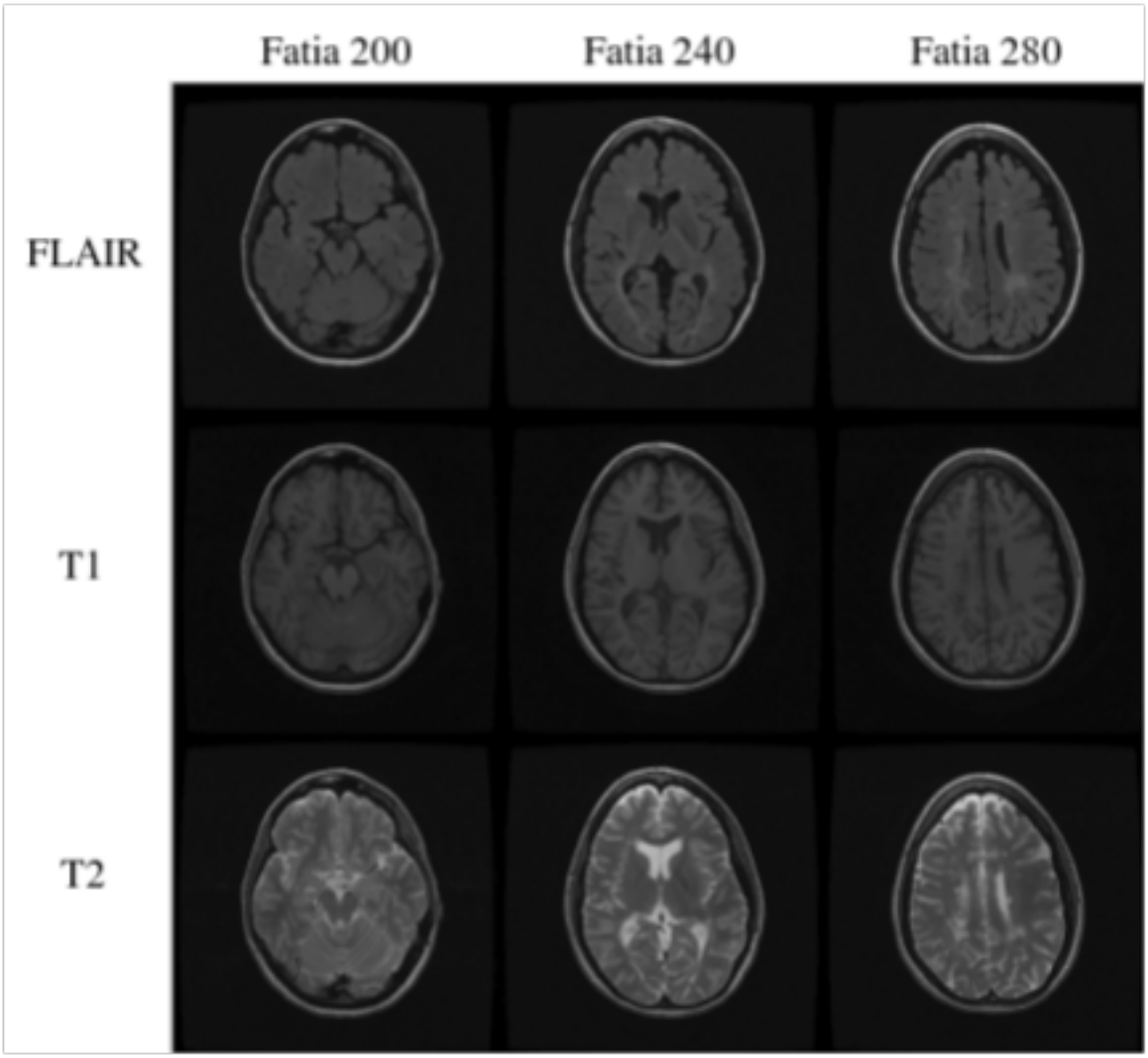

Figura 4 – Sequências de ressonância magnética FLAIR, T1 e T2. As linhas estão organizadas por tipo de sequência de aquisição. As colunas indicam o índice da fatia no eixo axial. Note as diferenças de contraste entre sequências.

2.3 SEGMENTAÇÃO DE LESÕES DE ESCLEROSE MÚLTIPLA

A segmentação manual de lesões de EM é demorada e sujeita a grande variabilidade intra e inter especialistas (GARCÍA-LORENZO et al., 2013). Como solução, a segmentação automática reduz a variabilidade, evita a interação humana e possibilita a segmentação de um grande número de imagens. Contudo, a segmentação automática de EM é dificultada por vários fatores relacionados as imagens de RM: demasiado ruído; escala de intensidade de pixel variável entre aquisições; tecidos diferentes com sobreposição de intensidades; presença de artefatos de aquisição como inomogeneidade do campo magnético e movimentação do paciente. Na Figura 5, casos de segmentação manual versus automática de lesões são apresentados.

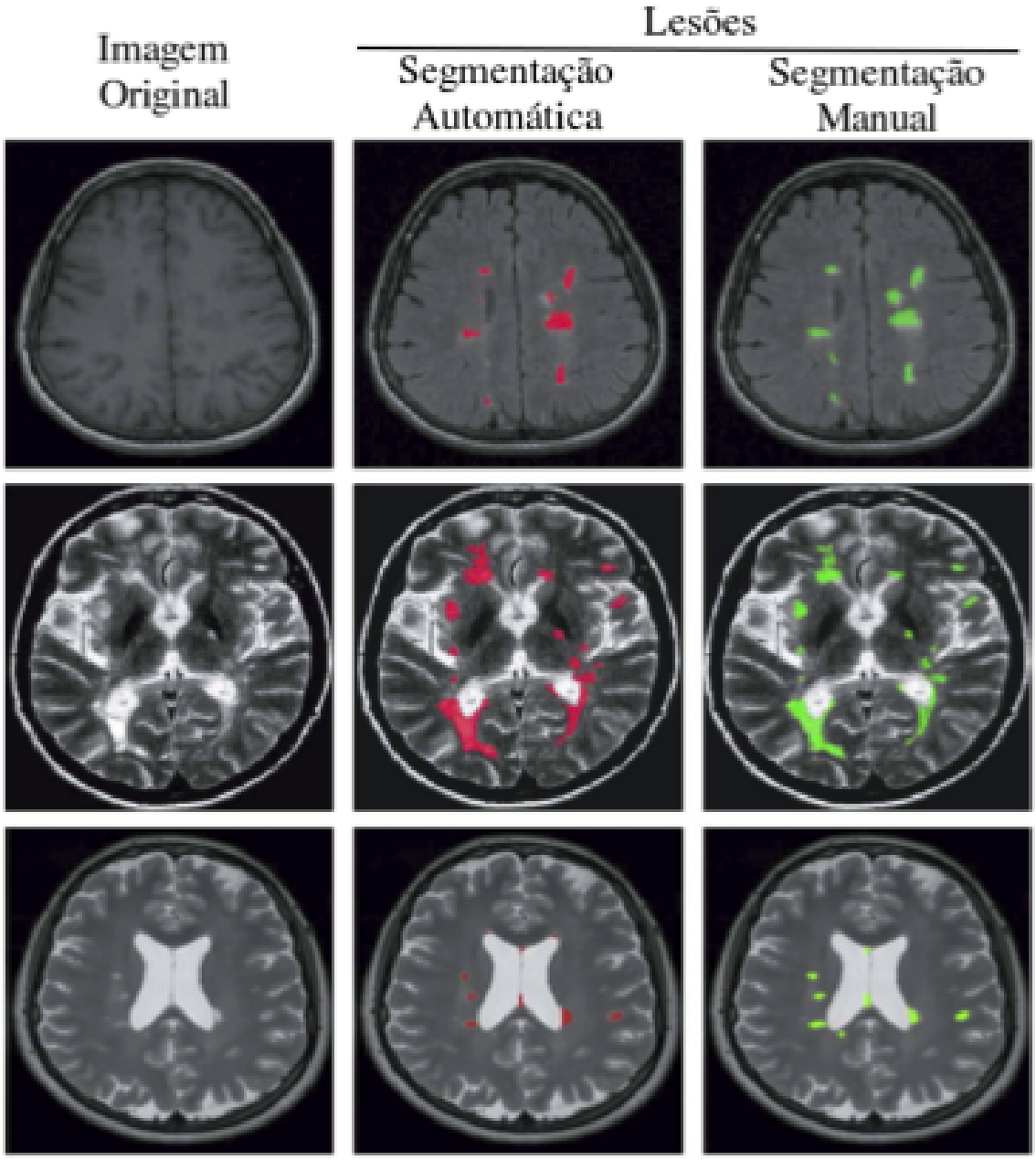

Figura 5 – Resultado da segmentação de lesões automática versus manual. A segmentação pelo método automático está em vermelho, enquanto a segmentação pelo método manual está em verde (adaptado de (LLADÓ et al., 2012)).

2.3.1 Segmentação Ideal: *Ground Truth*, Padrão-Ouro e Consenso

O treinamento e a validação de um algoritmo de segmentação de lesões requer a definição da segmentação ideal chamada de *ground truth*. As lesões de EM podem ser determinadas com exatidão através de exame histopatológico (BARKHOF et al., 1997), mas isto é pouco viável. Por outro lado, o imageamento por RM facilita a observação de

lesões *in-vivo*.

A partir das imagens de RM, as lesões demarcadas por especialistas constituem uma aproximação do *ground truth*, chamada padrão-ouro, que é mais viável, porém sujeita a uma série de problemas. Embora as imagens de RM apresentem alta especificidade (97%), as anomalias observáveis são apenas indícios das lesões reais ocorrendo nos tecidos (BARKHOF et al., 1997). Além disso, o estabelecimento do padrão-ouro é sujeito a variabilidades: intra-especialista, o mesmo especialista ao repetir a segmentação de uma imagem; inter-especialistas, diferentes especialistas ao segmentar a mesma imagem. Outros fatores que também contribuem para erros no padrão-ouro são diferentes equipamentos e protocolos de aquisição, além de variações normais ou patológicas entre pacientes. Nesta tese empregamos os termos *ground truth* e padrão-ouro de forma intercambiável.

Um modo de lidar com a variabilidade devida ao especialista é repetir a segmentação e gerar um consenso. Para tanto, é possível empregar o voto majoritário ou, de outro modo, o método proposto em (WARFIELD; ZOU; WELLS, 2004) que incorpora um modelo *a priori* da distribuição espacial das estruturas segmentadas.

2.3.2 Métricas de Desempenho de Segmentação

Uma métrica quantifica a similaridade ou diferença entre duas segmentações com base em atributos físicos. O atributo medido define o tipo da métrica: volume, sobreposição, forma, região, etc. As métricas servem para comparar segmentações entre especialistas humanos e, em particular, para comparar segmentação manual versus algorítmica. A escolha das métricas depende bastante de como um atributo medido impacta a aplicação final da segmentação. Em geral, métricas que quantificam similaridade e acurácia são do tipo quanto maior melhor, em oposição às métricas de distância e erro que são do tipo quanto menor melhor.

Em geral, uma única métrica não é capaz de fornecer toda a informação necessária para avaliar adequadamente a segmentação, sendo, portanto, necessário avaliar mais de uma métrica (GARCÍA-LORENZO et al., 2013; TAHA; HANBURY, 2015). A seguir são apresentadas as principais métricas de desempenho de segmentação.

2.3.2.1 Erro Quadrático Médio

O erro quadrático médio (*Mean Squared Error*, MSE) é uma medida da diferença ponto a ponto entre máscaras de segmentação. A medida é, usualmente, alvo de aprendizagem no treinamento de redes neurais (Equação 2.3). No entanto, não é comum medir diretamente o MSE para fins de validação da segmentação de EM.

2.3.2.2 Coeficiente de Similaridade de Dice

O coeficiente de similaridade de Dice (*Dice Similarity Coefficient*, DSC) é originalmente uma medida de associação entre conjuntos de espécies (DICE, 1945). Aplicado à segmentação de imagens, o DSC mede o volume sobreposto ponto a ponto entre duas segmentações e apresenta valores entre 0 e 1, quanto maior melhor. Esta métrica é sensível ao tamanho e ao volume total de lesão (GARCÍA-LORENZO et al., 2013). O cálculo do DSC é baseado em operações de conjuntos conforme:

$$\mathrm{DSC} = \frac{2|A \cap B|}{|A| + |B|}, \tag{2.6}$$

em que A e B são mascaras binárias de segmentações.

2.3.2.3 Diferença de Volume

Uma vez que a carga total de lesão (*total lesion load*, TLL) é frequentemente empregada como biomarcador em ensaios clínicos, métricas do volume total de *voxels* de lesão são usadas para avaliar a segmentação de EM (GARCÍA-LORENZO et al., 2013). Note que essas métricas não consideram sobreposição de qualquer tipo entre segmentação do algoritmo e segmentação manual. Em particular, a diferença de volume absoluta relativa (*relative absolute volume difference*, RAVD) computa a diferença entre volume da segmentação do algoritmo e volume da segmentação manual, em relação ao volume da segmentação manual, em valor absoluto, isto é:

$$\mathrm{RAVD} = \left| \frac{\mathrm{Vol}_A - \mathrm{Vol}_G}{\mathrm{Vol}_G} \right|, \tag{2.7}$$

em que Vol_A e Vol_G são os volumes das máscaras de segmentação do algortimo e do *ground-truth*, respectivamente. Quanto menor o RAVD, melhor a segmentação. Contudo, o valor ideal 0 pode ser obtido também para uma segmentação imperfeita, bastando para isto que os volumes entre segmentação algorítmica e manual sejam os mesmos (STYNER et al., 2008).

2.3.2.4 Métricas Por Lesão

Métricas por lesão envolvem a contagem de regiões contíguas sobrepostas entre segmentações. Assim, uma lesão corretamente detectada é a lesão apontada por um algoritmo que se sobrepõe com alguma lesão na segmentação manual. Note que isto é diferente das métricas de sobreposição que consideram o volume sobreposto ponto a ponto e não o número de regiões sobrepostas. Métricas por lesão são de particular interesse clínico, já que o número de lesões é critério de diagnóstico e progressão da EM (BARKHOF et al., 1997). A seguir são descritas algumas métricas por lesão.

A taxa verdadeiro positivo por lesão (lesion-wise true positive rate, LTPR) computa a fração de regiões corretamente detectadas por um algoritmo em relação ao número de lesões presentes na segmentação manual, isto é:

$$\text{LTPR} = \frac{\text{TP}_G}{M}, \tag{2.8}$$

em que TP_G é o número de lesões entre as M lesões do *ground truth* G que foram corretamente detectadas. Quanto maior o LTPR, melhor a segmentação. Note, entretanto, que é possível obter um LTPR de 100% mas espúrio, no caso em que o algoritmo aponta maior número de lesões do que há na segmentação manual, sendo importante observar outras métricas.

A taxa falso positivo por lesão (lesion-wise false positive rate, LFPR) é computada dividindo-se o número de lesões do algoritmo que não se sobrepõe com alguma lesão da segmentação manual pelo número de lesões da segmentação manual, isto é:

$$\text{LFPR} = \frac{\text{FP}_A}{M}, \tag{2.9}$$

em que FP_A é o número de lesões do algoritmo A que não foram detectadas corretamente entre as M lesões do *ground truth*. Quanto menor o LFPR, melhor a segmentação. Uma desvantagem é que uma sobressegmentação pode resultar em valores baixo para esta métrica, desde que todas as lesões no *ground-truth* tenham sido sobrepostas por uma ou mais lesões do algoritmo (STYNER et al., 2008). Além disso, uma segmentação do algoritmo sem lesões apontadas resulta em um LFPR de 0% que é, claramente, espúrio.

O valor preditivo positivo por lesão (lesion-wise positive predictive value, LPPV) mede a fração de regiões corretamente detectadas em relação ao número total de lesões preditas pelo algoritmo, isto é:

$$\text{LPPV} = \frac{\text{TP}_A}{N}, \tag{2.10}$$

em que TP_A é o número de lesões entre as N lesões do algoritmo A que foram corretamente detectadas. Um LPPV de 100% representa uma segmentação perfeita. No entanto, um valor máximo enganoso pode ser obtido se a segmentação do algoritmo contém um menor número de lesões do que há na segmentação manual.

O *F1-score* agrega o LTPR e o LPPV, como tentativa de contornar os problemas de usar uma única métrica (COMMOWICK et al., 2018). Além disso, essa métrica é um análogo por lesão do índice de Dice ponto a ponto, indo de 0 a 1 quanto maior melhor. O cálculo do F1-score é dado pela seguinte equação:

$$\text{F1-score} = 2\frac{\text{LTPR} \times \text{LPPV}}{\text{LTPR} + \text{LPPV}}, \tag{2.11}$$

As métricas por lesão são menos sujeitas a variabilidade da segmentação manual, mas não capturam a informação de acurácia nas bordas da lesão que pode estar subsegmentada ou sobressegmentada. Além disso, como apontado em (GARCÍA-LORENZO et al., 2013), o

número de verdadeiros positivos por região pode ser dúbio. Se, por exemplo, duas lesões da segmentação do algoritmo se sobrepõe com uma lesão do padrão-ouro. Então, o número de acertos por lesão pode referir-se as duas lesões preditas pelo algoritmo ou a uma lesão apontada no padrão-ouro.

2.3.2.5 Distância Entre Superfícies

Uma métrica baseada em distância entre superfícies considera as distâncias entre os pontos do contorno das lesões na segmentação manual em relação a segmentação do algoritmo e vice-versa. As distâncias entre as superfícies de regiões servem para comparar a forma das lesões. Do ponto de vista clínico, lesões com forma bem-definida são achados frequentes e interesse primário para o estudo da EM (BARKHOF et al., 1997; GARCÍA-LORENZO et al., 2013). A distância média simétrica entre superfícies (average symmetric surface distance, ASSD) é calculada conforme:

$$\text{ASSD} = \frac{\sum_{a \in A} \min \text{dist}(a, B) + \sum_{b \in B} \min \text{dist}(b, A)}{|A| + |B|}, \tag{2.12}$$

em que A e B são superfícies, $\text{dist}(a, B)$ é a menor distância entre cada ponto $a \in A$ até a superfície B e, analogamente, $\text{dist}(b, A)$ é a menor distância entre cada ponto $b \in B$ até a superfície B.

Ao contrário de medidas baseadas em sobreposição, métricas baseadas em distância de superfície fornecem valores não-nulos mesmo quando não há sobreposição de regiões e, por isso, tendem a ser mais sensíveis a segmentação de baixa qualidade (TAHA; HANBURY, 2015). Um problema com as medidas de distância de superfície é que a interpretação do resultado pode tornar-se difícil ao comparar contornos de múltiplas lesões.

3 REVISÃO DA LITERATURA

Neste capítulo conduzimos uma revisão crítica da literatura relacionada ao problema de pesquisa. Primeiro, na Seção 3.1, tratamos da função objetivo específica para segmentação de esclerose múltipla com multilayer perceptron. Depois, na Seção 3.2, tratamos da máquina de aprendizado extremo convolucional para reconhecimento de dígitos.

3.1 FUNÇÃO OBJETIVO PARA SEGMENTAÇÃO DE ESCLEROSE MÚLTIPLA

O treinamento de um classificador neural supervisionado pode ser visto como um problema de otimização (HAYKIN, 2008, Sec. 4.16). Analogamente, o treinamento de um método de segmentação de lesões de EM, não necessariamente baseado em rede neural, pode ser visto como problema de otimização. Deste modo, a tarefa de otimização do método de segmentação consiste em encontrar um conjunto de parâmetros, que otimiza uma função objetivo, cujo valor é amostrado sobre o conjunto de treinamento. Após a etapa de otimização, o método segmentação é submetido a uma etapa de avaliação que consiste em computar métricas sobre o conjunto de dados de teste. Nesta Seção, revemos aspectos da otimização e da avaliação de métodos segmentação de lesões de EM.

Esta seção está dividida em subseções organizadas como segue. Na Subseção 3.1.1, apontamos o problema da divergência entre função objetivo e métricas de segmentação de EM. Em seguida, na Subseção 3.1.2, discutimos a inviabilidade de empregar métricas de segmentação diretamente como função objetivo no treinamento iterativo. Por fim, apontamos possíveis vantagens e desvantagens de substituir métricas tridimensionais por métricas bidimensionais como função objetivo para segmentação de EM, na Subseção 3.1.3.

3.1.1 Divergência entre Função Objetivo e Métricas de Segmentação de Esclerose Múltipla

Frequentemente, a função objetivo na otimização do método de segmentação *diverge* (ou difere) das métricas de segmentação na avaliação, ao não coincidirem a função objetivo às métricas de segmentação, o que pode levar a resultados subótimos (LECOEUR; FERRÉ; BARILLOT, 2009). O Quadro 1 compara função objetivo na otimização com as métricas de segmentação na avaliação. Observando a primeira e a segunda coluna, nos três primeiros trabalhos, os autores não coincidem a função objetivo com as métricas de segmentação de EM (GUIZARD et al., 2015; TOMAS-FERNANDEZ; WARFIELD, 2015; SOUPLET et al., 2008).

Há trabalhos que parcialmente coincidem função objetivo e métricas de segmentação de EM. Por exemplo, em (BROSCH et al., 2016), ao treinar um tipo de CNN, a função objetivo teve o erro quadrático ponderado por sensibilidade (TPR) e especificidade (1−FPR).

Quadro 1 – Função Objetivo versus Métrica Usadas na Segmentação de EM

Função Objetivo (Otimização)	Métricas de Segmentação (Avaliação)	Objetivo Coincide com Métricas	Referência
Rotation-Invariant Distance	DSC, TPR, LTPR, PPV, ...	Não	(GUIZARD et al., 2015)
Trimmed Likelihood, *Energy*	LTPR, LFPR, DSC, PPV, ...	Não	(TOMAS-FERNANDEZ; WARFIELD, 2015)
Log-likelihood	LTPR, LFPR, RAVD, ASSD	Não	(SOUPLET et al., 2008)
MSE *Weighted* by TPR & TNR	TPR, TNR, LTPR, LFPR	Parcialmente	(BROSCH et al., 2016)
Energy, ROC, Jaccard	Jaccard, DSC, TPR, FPR	Parcialmente	(ZHAN et al., 2015)
Trimmed Likelihood, DSC	DSC, TLL, Se, Sp	Parcialmente	(GARCÍA-LORENZO et al., 2011)
DSC	DSC e F1-score	Parcialmente	(GALASSI et al., 2019)
DSC, LTPR, LPPV	DSC, LTPR, LPPV	Sim	(ROURA et al., 2015)

A motivação para essa função objetivo foi contornar o problema do predomínio da classe majoritária na saída do classificador, devido ao desbalanceamento de classes que ocorre em bases de dados como as de segmentação de EM. Apesar de objetivar sensibilidade e especificidade no treinamento da CNN, outras métricas de segmentação foram consideradas apenas na avaliação dos resultados.

Em outro trabalho (ZHAN et al., 2015), os autores propõem um método em duas etapas: na primeira, a segmentação é realizada por limiarização do z-score de intensidade, em que a métrica ou indice de Jaccard é usada para obter o valor de limiar ótimo; em seguida, a segmentação inicial serve como conhecimento *a priori* para um método de segmentação por *level-set*. O método em duas etapas com otimização pelo índice de Jaccard teve desempenho de segmentação superior a usar apenas o método *level set*, empregando essa mesma métrica para avaliação. As métricas DSC, TPR e FPR foram empregadas apenas para comparação do desempenho de segmentação com outros métodos.

Em trabalho recente, Galassi et al. (2019) emprega o DSC generalizado como função objetivo específica para segmentação no treinamento do modelo de Valverde et al. (2017) com duas CNNs em cascata. Para avaliação dos resultados de segmentação de lesões de EM, além do DSC, o trabalho usa a métrica por lesão F1-score. Apesar de empregar apenas o DSC como função objetivo, o trabalho reportou um desempenho de segmentação que representa um ligeiro avanço do estado-da-arte sobre o conjunto de dados MICCAI2016. No entanto, tal abordagem permanece aquém do desempenho humano nesse mesmo conjunto de dados (COMMOWICK et al., 2018). Um desempenho de segmentação melhor poderia

ser obtido se a definição da função objetivo considerasse também o F1-score. Note que a rede neural empregada tem centenas de milhares de parâmetros ajustados por descida de gradiente (VALVERDE et al., 2017), o que está associado a um longo tempo de treinamento (algo indesejável conforme Seção 1.4).

Uma das exceções a escassez de algoritmos explicitamente otimizados para métricas de segmentação esta em (ROURA et al., 2015). Nesse artigo, o algoritmo parte da segmentação dos tecidos encefálicos em substância branca, substância cinzenta e fluído cerebrospinal. Em seguida, as lesões são segmentadas como *outliers* na distribuição de intensidades da substância cinzenta com base em um limiar. Por fim, duas iterações de refinamento são conduzida para remover falso positivos, também através limiares. Os valores dos limiares são obtidos por otimização exaustiva das métricas DSC, LTPR, LPPV, que são também empregadas para avaliar os resultados. Uma limitação dessa abordagem está na otimização exaustiva, viável para um pequeno número de parâmetros, mas potencialmente não aplicável para redes neurais MLP com cerca de centenas de parâmetros.

Assim, os algoritmos de segmentação de EM, tipicamente, tem objetivo de treinamento que difere das métricas usadas para avaliar a segmentação, o que pode ser uma limitação, já que as métricas usadas para avaliar a segmentação de EM representam critérios de interesse clínico como volume, forma e contagem de lesões (STYNER et al., 2008; BARKHOF et al., 1997). Trabalhos anteriores demonstram que coincidir métricas de segmentação de EM com o objetivo de treinamento pode melhorar o desempenho de segmentação medido, por exemplo (LECOEUR; FERRÉ; BARILLOT, 2009). Especificamente, métodos baseados em redes neurais tem como função objetivo medidas do erro de segmentação ponto-a-ponto (ZIJDENBOS; FORGHANI; EVANS, 2002; CERASA et al., 2012; VAIDYA et al., 2015). Todavia, até onde sabemos, não há abordagem anterior, baseada em redes neurais MLP, que coincide função objetivo com métricas de segmentação de EM.

3.1.2 Inviabilidade das Métricas de Segmentação como Função Objetivo para Treinamento Iterativo

As métricas de segmentação podem ser computacionalmente custosas para serem usadas diretamente como função objetivo. O tempo de cálculo das métricas geralmente não é problema na avaliação da segmentação, em que cada métrica é calculada poucas vezes e na ordem de segundos. No entanto, uma métrica de segmentação como função objetivo que leva segundos para ser obtida pode ser um sério obstáculo para otimização iterativa, porque o objetivo é computado muitas vezes. Além disso, métricas baseadas em contagem de lesões dependem do método de componentes conectados para obtenção de regiões contíguas. Computar componentes conectados também pode levar segundos em imagens 3D (HE; CHAO; SUZUKI, 2011).

Como exemplo para verificar que a otimização iterativa de métricas de segmentação pode ser pouco viável, suponha que computar um conjunto de medidas gaste 1 minuto

para cada um de 10 pacientes. Neste caso, em uma única execução de um algoritmo de treinamento com 1000 iterações, o procedimento levaria quase uma semana (6,9 dias).

3.1.3 Substituição de Métricas Tridimensionais por Métricas por Fatia como Função Objetivo para Segmentação de Esclerose Múltipla

O tempo gasto por uma métrica de segmentação de EM pode ser reduzido se, ao invés de computar em três dimensões, realizamos o cálculo por fatia. Componentes conectados podem ser computados em milissegundos para imagens 2D (HE et al., 2009). Em geral, métricas de segmentação podem ser computadas em frações de segundos com máscaras binárias 2D. Voltando ao exemplo da Seção 3.1.2, se as métricas de segmentação como objetivo forem computadas em 100 milissegundos para cada um de 10 pacientes, um treinamento com 10.000 iterações levaria apenas 17 minutos, ao invés de quase 1 semana do exemplo mencionado (Tabela 1).

Tabela 1 – Tempo de Treinamento com 1000 Computações das Métricas de Segmentação para 10 Pacientes

Métricas como Função Objetivo	Tempo de Cálculo por Paciente (s)	Tempo Total de Treinamento (s)
3D	60	600.000 ($\sim$ 1 semana)
2D	0,1	1.000 ($\sim$ 17 minutos)

A aceleração obtida ao substituir métricas tridimensionais por bidimensionais pode ser bastante impactante: proporcional ao número N de elementos considerados para cálculo na imagem 3D original, vide Tabela 2. Note, porém, que a otimização por função objetivo substituta implica em perda de desempenho de segmentação (JIN, 2011). Por isso, é importante avaliar a fidelidade (correlação) da métrica tridimensional com a métrica por fatia.

Tabela 2 – Complexidade Computacional e Aceleração Teórica em Métricas 3D versus 2D

Métricas	Complexidade		Aceleração Teórica	Relativo a
	3D	2D		
Contagem de Lesões (Componentes Conectados) (HE; CHAO; SUZUKI, 2011)	$O(N^3)$	$O(N^2)$	N	N Regiões
Distância de Superfície (RUSS; KOCH; LITTLE, 2005)	$O(N^2)$	$O(N)$	N	N Pontos de 2 Superfícies
Volume	$O(N^3)$	$O(N^2)$	N	N Pontos de Volume

3.2 MÁQUINA DE APRENDIZADO EXTREMO CONVOLUCIONAL

Nesta seção, a apresentação da revisão da literatura está dividida em subseções como segue. Na Subseção 3.2.1 revisamos redes neurais convolucionais de treinamento rápido e contrastamos com nossa CELM proposta. Já na Seção 3.2.2, tratamos de filtros convolucionais. Enquanto, na Subseção 3.2.3, abordamos a validação empírica de modelos de erro de generalização envolvendo CELMs.

3.2.1 Rede Neural Convolucional de Treinamento Rápido

Em um dos primeiros trabalhos a combinar extração convolucional de características e treinamento rápido de pesos de saída por mínimos quadrados, a *Extreme Learning Machime Local Receptive Field* (ELM-LRF) foi proposta em (HUANG et al., 2015b). Na ELM-LRF, o único estágio de extração de características foi composto por uma camada convolucional e uma camada de pooling. Filtros convolucionais aleatórios ortogonais foram empregados como campos receptivos locais. No classificador, apenas uma camada de pesos de saída, não há camada escondida com pesos aleatórios. Aplicada ao reconhecimento de objetos, através do conjunto de dados NORB, essa rede neural obteve erro competitivo comparada às abordagens prévias.

Uma extensão da ELM-LRF, a rede ELM com *Multi-scale Local Receptive Field* (ELM-MSLRF) emprega filtros convolutivos de tamanhos diferentes em um único estágio de extração de características, a fim de capturar padrões em diferentes escalas (HUANG et al., 2017a). Assim como em (HUANG et al., 2015b), o estágio de extração contou com camadas de convolução e pooling. Mas diferente do trabalho anterior, o estágio de classificação foi composto por uma camada escondida aleatória, além da camada de saída. Outra diferença, os filtros convolutivos foram aleatórios simples na ELM-MSLRF. Sobre a base MNIST, o

algoritmo multi-escala foi superior em acurácia e com tempo de treinamento ligeiramente maior em relação ao equivalente mono-escala.

Uma CELM profunda foi proposta e aplicada ao reconhecimento de dígitos manuscritos em (PANG; YANG, 2016). Este trabalho foi pioneiro em avaliar o aumento da profundidade em redes convolucionais rápidas. Os autores obtiveram que a profundidade ideal pode variar com o conjunto de dados. Uma das limitações do estudo é que apenas filtros convolucionais aleatórios ortogonalizados foram avaliados. Além disso, no estágio de classificação, apenas uma camada de pesos de saída foi empregada.

Na rede CELM com *kernel* (*Convolutional Extreme Learning Machine with Kernel*, CKELM), em (DING; GUO; HOU, 2017), a extração de características envolve quatro camadas encadeadas com filtros convolucionais aleatórios, equivalendo a dois estágios de extração. O estágio de classificação consiste de uma ELM com kernel Gaussiano composta por uma camada oculta com pesos aleatórios e uma camada com pesos por mínimos quadrados. Na aplicação em reconhecimento de dígitos, a acurácia foi considerada melhor que de outras redes rápidas não-convolucionais. O tempo de treinamento da CKELM foi apenas ligeiramente maior que o de uma rede KELM, isto é, sem os estágios de extração de características.

Em (MCDONNELL; VLADUSICH, 2015), uma rede convolucional rasa com treinamento rápido foi proposta para tarefas de classificação de imagens. Diferindo de outras redes convolucionais rápidas que contam apenas com filtros aleatórios, o método diversificou usando filtros recorte, heurístico e de aprendizado transferido. Tal rede neural resultou em acurácia competitiva, mas com tempo de treinamento muito menor que abordagens baseadas em descida de gradiente. Recentemente, usando a mesma arquitetura, uma versão semi-supervisionada da rede convolucional rápida foi proposta com filtros aprendidos via K-means (YOUSEFI-AZAR; MCDONNELL, 2017). Os resultados confirmaram que, mesmo com filtros aprendidos, essa rede neural apresenta rápido treinamento, mantendo acurácia competitiva. Além disso, o filtro K-means mostrou maior poder discriminativo comparado ao filtro aleatório. Apesar do relativo sucesso, essas CELMs apresentam algumas limitações. Na arquitetura, essas redes contam com apenas um estágio de extração de características, o que pode limitar sua acurácia. Além disso, ainda que nesses estudos as CELMs usem diferentes tipos de filtros convolucionais, eles não conduziram uma comparação estatística de filtros e combinações de filtros.

Em resumo, os trabalhos anteriores apresentam limitações em três pontos, que nós contrastamos com nossas propostas a seguir. Primeiro, a maioria das redes convolucionais rápidas são baseadas em filtros aleatórios (HUANG et al., 2015b; PANG; YANG, 2016; HUANG et al., 2017a; DING; GUO; HOU, 2017). Mas, filtros de diferentes tipos podem resultar em diversidade de características extraídas, aumentando o poder discriminativo. Segundo, a profundidade é muitas vezes limitada, com somente uma camada convolucional (MCDONNELL; VLADUSICH, 2015; HUANG et al., 2015b; HUANG et al., 2017a; YOUSEFI-AZAR;

MCDONNELL, 2017). Porém, múltiplos estágios de extração de características formados por convolução e pooling podem extrair informação em diferentes níveis/escalas. Terceiro, na classificação, as características extraídas são ligadas diretamente a camada de saída em três trabalhos (HUANG et al., 2015b; HUANG et al., 2017a; PANG; YANG, 2016). Em contraste, atuando entre o último estágio de extração de características e camada de saída, a camada oculta consiste em uma projeção aleatória que permite controlar o tamanho do espaço de características através do número neurônios.

Em contraste com as abordagens acima mencionadas, a CELM proposta reúne características que não estavam presentes antes em uma única rede convolucional rápida: (1) combinação de quatro tipos de filtros convolucionais; (2) dois estágios de extração de características; (3) duas camadas no estágio de classificação, incluindo uma camada oculta com pesos aleatórios restritos. Na Tabela 3.2.1, comparamos as CELMs anteriores com nossa proposta.

Tabela 3 – Arquiteturas de Redes Convolucionais de Treinamento Rápido

Referência	Tipos de Filtro	Estágios de Extração de Características	Camadas no Estágio de Classificação
(HUANG et al., 2015b)	Aleatório	1	1
(HUANG et al., 2017a)	Aleatório	1	1[a]
(PANG; YANG, 2016)	Aleatório	2-3[b]	1
(DING; GUO; HOU, 2017)	Aleatório	2	2
(MCDONNELL; VLADUSICH, 2015)	Heurístico, Recorte, Transferido	1	2
(YOUSEFI-AZAR; MCDONNELL, 2017)	K-means	1	2
Presente Trabalho	Aleatório, PCA, Recorte, Gabor	2	2

[a]Como em (PANG; YANG, 2016), concatenação de vetores não é considerada uma camada.
[b]Número de etapas de extração para os menores erros obtidos.

3.2.2 Filtros Convolucionais

Apesar de promissoras, as redes neurais convolucionais rápidas apresentam alguns desafios. Uma questão é como definir os bancos de filtros convolucionais uma vez que não são finamente ajustados por retropropagação de gradientes. Em CNNs com filtros alea-

tórios e pesos do classificador ajustados por gradiente, a acurácia é surpreendentemente não muito pior que a da mesma rede com todos os pesos ajustados (JARRETT et al., 2009; SAXE et al., 2011). Essa tem sido uma das motivações para boa parte das redes convolucionais rápidas empregarem filtros aleatórios (HUANG et al., 2015b; DING; GUO; HOU, 2017). Embora os filtros aleatórios resultem em acurácia respeitável, eles estão longe de serem a melhor opção (YOUSEFI-AZAR; MCDONNELL, 2017). Além disso, há uma variedade de tipos de filtros convolutivos que não foram combinados e avaliados em redes convolucionais rápidas.

3.2.3 Validação Empírica do Erro de Generalização

Em redes convolucionals rápidas, trabalhos que validam modelos teóricos do erro de generalização contra dados empíricos são escassos. Em especial, o erro de generalização *conjuntamente* em função do número de exemplos e do número de neurônios escondidos permanece pouco estudado. Os trabalhos analisam o erro de generalização somente em função do número de neurônios (DING; GUO; HOU, 2017; MCDONNELL; VLADUSICH, 2015; YOUSEFI-AZAR; MCDONNELL, 2017), ou do número de exemplos (HUANG et al., 2017a; PANG; YANG, 2016).

O teorema de Rahimi e Recht estabelece um limite superior para o erro de teste em função do número de exemplos de treinamento e do número de neurônios escondidos (RAHIMI; RECHT, 2009). As condições do teorema de Rahimi-Retch definem uma rede com uma camada escondida de pesos aleatórios e uma camada de saída com pesos por mínimos quadrados, sendo compatível com o estágio de classificação de redes convolucionais rápidas tais como as que foram propostas em (DING; GUO; HOU, 2017; MCDONNELL; VLADUSICH, 2015; YOUSEFI-AZAR; MCDONNELL, 2017).

Dois aspectos apontados por Krizhevsky, Sutskever e Hinton (2012) como possíveis fatores para o baixo erro de generalização obtido com redes neurais profundas são abordados no teorema de Rahimi-Retch: o tamanho do conjunto de dados e a capacidade de representação do modelo. Este último aspecto é representado no teorema Rahimi-Retch pelo número de neurônios escondidos, o que determina o número de parâmetros (pesos) ajustáveis na camada oculta. Uma possível interpretação do teorema de Rahimi-Retch é que o erro de generalização tende a decrescer conforme crescem, *ao mesmo tempo*, o tamanho do conjunto de treinamento e número de pesos ocultos.

Na formulação do teorema de Rahimi-Retch, os autores apresentaram o algoritmo de treinamento chamado pias de cozinha aleatórias (*Random Kitchen Sinks*, (RKS)) representando uma classe de redes neurais de treinamento rápido anteriores para as quais este teorema é adequado (RAHIMI; RECHT, 2009). RKS e ELM são intimamente relacionados. O RKS pode usar bases de Fourier como função ativação (ou característica) na camada oculta (RAHIMI; RECHT, 2007), assim como pode ocorrer na ELM (HAN; HUANG, 2006). No entanto, o teorema de Rahimi-Reich não requer bases de Fourier e assume apenas que

a função de ativação deve ser limitada. Outros limites de generalização para redes com camada oculta aleatória foram fornecidos em estudos anteriores (LIU et al., 2015; BACH, 2017; RUDI; ROSASCO, 2017). Nesta tese, empregamos o modelo de Rahimi-Retch devido à sua aplicação simples e direta, além da capacidade de explicar os dados observados.

3.2.3.1 O conjunto de dados EMNIST

O estudo empírico do erro de generalização assintótico requer um grande número de exemplos. No lado experimental, o conjunto de dados *extended MNIST* (EMNIST) prove um número maior de dígitos que o típico MNIST (COHEN et al., 2017). Ambos EMNIST e MNIST são derivados do NIST Special Database 19. O EMNIST é conveniente porque estende o MNIST usando o mesmo formato de dados. No entanto, o MNIST é um benchmark tradicional e amplamente adotado. As redes convolucionais rápidas são frequentemente avaliadas no reconhecimento de dígitos com o conjunto de dados MNIST (HUANG et al., 2017a; PANG; YANG, 2016; DING; GUO; HOU, 2017; MCDONNELL; VLADUSICH, 2015).

4 OBJETIVOS DA PESQUISA

O presente capítulo parte das limitações apontadas na revisão da literatura, passando pelas questões (Seção 4.1) e hipóteses de pesquisa (Seção 4.2). Por fim, concluímos o capítulo definindo os objetivos (Seção 4.3).

4.1 QUESTÕES

A fim de tentar superar as limitações apontadas pela revisão da literatura derivamos questões de pesquisa. No Capítulo 3, nossa revisão da literatura identificou limitações relacionadas às funções objetivo em métodos de segmentação de EM e às máquinas de aprendizado extremo convolucionais. As Questões 1 e 2 tratam de métricas de segmentação como função objetivo específica no MLP-PSO para segmentação de lesões de EM. Ambas as questões que seguem lidam com CELM, comparando bancos de filtros (Questão 3), ou buscando um modelo do erro de generalização de uso prático (Questão 4). As questões de pesquisa são listadas a seguir.

1. **Métricas de segmentação de EM como função objetivo específica no MLP-PSO melhoram o desempenho de segmentação de EM observado?**
2. **O algoritmo de segmentação de EM usando MLP-PSO com métricas de segmentação na função objetivo compara-se ao estado-da-arte?**
3. **Qual banco de filtros convolucionais resulta em menor error da CELM para reconhecimento de dígitos?**
4. **Qual modelo de erro de generalização explica o erro empírico da CELM para reconhecimento de dígitos?**

4.2 HIPÓTESES

Nesta seção, cada hipótese fornecem uma resposta provisória, a ser averiguada, para uma questão de pesquisa. Seguimos a definição de hipótese de pesquisa como sendo uma afirmação, simples e testável. As hipóteses testadas nesta tese são declaradas a seguir.

1. **Métricas de segmentação de EM como função objetivo específica resulta em desempenho de segmentação melhor do que a função objetivo inespecífica.**

   Métricas de segmentação por fatia, como função objetivo específica, podem resultar em melhor desempenho de segmentação do que com uma função objetivo inespecífica

representada pelo MSE. Para verificar essa hipótese, empregamos o conjunto de dados MICCAI2008.

2. **O algoritmo de segmentação usando MLP-PSO com métricas de segmentação na função objetivo resulta em desempenho de segmentação equivalente ao estado-da-arte.**

   O algoritmo de segmentação usando MLP-PSO com métricas de segmentação na função objetivo pode resultar em desempenho de segmentação equivalente a outros métodos de segmentação de EM do estado-da-arte. Para testar essa hipótese, consideramos os conjuntos de dados MICCAI2008 e MICCAI2016 como referências.

3. **Filtros combinados resultam em menor erro do que filtros puros na CELM para reconhecimento de dígitos**

   A combinação de filtros convolucionais de diferentes tipos, em estágios de extração de características, pode resultar em maior poder discriminativo da CELM para reconhecimento de dígitos.

4. **O modelo de erro de generalização baseado no teorema de Rahimi-Retch concorda e prediz o erro empírico na CELM para reconhecimento de dígitos**

   O erro empírico é ajustado contra um modelo do erro de generalização. Então, o modelo ajustado pode ser avaliado quanto a concordância e capacidade de predição do erro de teste da CELM em reconhecimento de dígitos.

4.3 OBJETIVOS

Nesta seção definimos as ações para testar as hipóteses levantadas na seção anterior. A seguir, definimos estas ações, de modo amplo em Objetivo Geral e de modo pontual em Objetivos Específicos.

4.3.1 Objetivo Geral

O objetivo geral desta tese é investigar classificadores baseados em redes neurais, tendo idealmente correção equivalente ao humano e treinamento rápido, para aplicação em segmentação de lesões de EM, bem como em reconhecimento de dígitos.

4.3.2 Objetivos Específicos

Os objetivos específicos estão declarados abaixo.

1. **Desenvolver funções objetivo específica e inespecífica, algoritmo de treinamento MLP-PSO e algoritmo de segmentação de lesões de EM.**

2. **Comparar o uso da função objetivo específica com o uso da função objetivo inespecífica, e também com o estado-da-arte em segmentação de EM, empregando as bases de dados MICCAI2008 e MICCAI2016 como referência.**

3. **Desenvolver bancos de filtros convolucionais, modelo de erro generalização e arquitetura de CELM para reconhecimento de dígitos.**

4. **Avaliar a CELM, comparando bancos de filtros combinados e puros na CELM, ajustando e predizendo o erro empírico com o modelo de erro de generalização, empregando o conjunto de dados de reconhecimento de dígitos EMNIST.**

5 METODOLOGIA

Este capítulo detalha os métodos desenvolvidos em redes neurais artificiais e aplicados em tarefas de classificação de padrões de imagens. A Seção 5.1 aborda os métodos relacionados à função objetivo específica para a tarefa de segmentação de lesões de esclerose múltipla. Enquanto, a Seção 5.2 descreve os métodos que dizem respeito à máquina de aprendizado extremo convolucional para a tarefa de reconhecimento de dígitos.

5.1 FUNÇÃO OBJETIVO PARA SEGMENTAÇÃO DE ESCLEROSE MÚLTIPLA

Nesta seção descrevemos o método de segmentação de EM desenvolvido com ênfase na função objetivo específica proposta. Iniciamos por uma descrição de alto nível das principais etapas de processamento, na Seção 5.1.2. Em seguida, as Seções 5.1.3 e 5.1.4 descrevem as bases de dados empregadas. Depois, a Seção 5.1.5 trata do pré-processamento, enquanto a Seção 5.1.6 lida com a amostragem. Então, a Seção 5.1.7 contém o detalhamento do classificador, enfatizando a função objetivo específica para segmentação de esclerose múltipla. A Seção 5.1.8 descreve as etapas de pós-processamento, encerrando o capítulo.

5.1.1 Ambiente de *Hardware* e *Software*

Os experimentos foram executados em uma máquina com dois processadores *Intel Xeon* de 2,4 GHz de 6 núcleos cada e 16 GB de memória RAM. As rotinas do MLP-PSO foram implementadas no *software* MATLAB. Os scripts de pré-processamento foram codificados em *GNU bash*. O sistema operacional empregado foi OS X 10.10 (Yosemite).

5.1.2 Visão Geral

O método de segmentação de esclerose múltipla pode envolver várias etapas de processamento. A Figura 6 exibe as principais etapas de processamento do método de segmentação de lesões de esclerose múltipla.

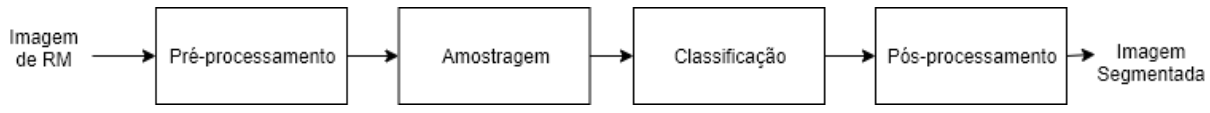

Figura 6 – Etapas do método de segmentação de lesões de esclerose múltipla.

5.1.3 Conjunto de Dados MICCAI2008

A *MS Lesion Segmentation Challenge 2008* é uma competição de algoritmos de segmentação de EM que foi iniciada durante a conferência internacional Medical Image Computing and Computer Assisted Intervention of 2008 (MICCAI2008). O *website*[1] da competição permance fornecendo a base de dados e recebendo novas submissões de segmentações. Para cada paciente no conjunto de dados, três imagens foram adquiridas com diferentes seqüências de RM denominadas: T1w, T2w e FLAIR. Todas as imagens foram corregistradas e reamostradas para uma matriz de 512x512x512, correspondendo a um tamanho de voxel de 0,5x0,5x0,5mm. A segmentação foi feita por especialistas humanos, atribuindo a cada voxel os valores: 1, para lesão; 0, caso contrário.

Na competição, as imagens são divididas em dois conjuntos: *treinamento* e *teste*. O conjunto de treinamento possui *ground-truth* público e consiste em 20 casos, enquanto o conjunto de testes tem 23 pacientes e *ground-truth* mantido em sigilo.

Para avaliação, o site da competição atribui uma pontuação baseada na segmentação realizada sob o conjunto de imagens de teste. Esta pontuação de avaliação compreende quatro métricas calculadas em 3D: diferença de volume absoluta relativa (*relative absolute volume difference*, RAVD); distância de superfície simétrica média (*average symmetric surface distance*, ASSD); taxa verdadeiro-positivo por lesão (*lesion-wise true positive rate*, LTPR); taxa de falso positivo por lesão (*lesion-wise false positive rate*, LFPR). A pontuação final (3D) é uma média entre as quatro métricas. Cada uma das métricas é transformada para ficar no intervalo [0, 100] e para ser quanto maior, melhor. Uma pontuação de 90 é equivalente a segmentação esperada de um especialista humano independente.

5.1.4 Conjunto de Dados MICCAI2016

A *MS Lesion Segmentation Challenge 2016* é outra competição de algoritmos de segmentação de EM que teve início na conferência MICCAI2016[2]. Em comparação com a competição MICCAI2008, além de um conjunto de dados e métricas de avaliação, essa competição forneceu uma infra-estrutura computacional comum para executar algoritmos competidores. O conjunto de dados é composto por 53 pacientes (15 para treinamento, 38 para teste). Os dados foram adquiridos em aparelhos de RM de 1,5T ou 3T, em 4 centros médicos distintos. Para cada exame de um paciente, o conjunto de dados contém segmentações de 7 especialistas como *ground-truth*, além de imagens de 5 seqüências de RM nomeadas como segue: 3D FLAIR, 3D T1-w, 3D T1-w GADO, 2D DP e 2D T2-w. Os algoritmos para cálculo das métricas de avaliação usadas no MICCAI2016 estão disponíveis integrando o *sotfware* Anima[3].

[1] Em <http://www.ia.unc.edu/MSseg>, último acesso em 13/12/2019.

[2] As competições de segmentação de EM não tem uma periodicidade definida, dependendo de iniciativas de grupos de pesquisadores para realização.

[3] Em <https://github.com/Inria-Visages/Anima-Public/wiki/Segmentation-tools>, último acesso em 13/12/2019.

Para avaliação, um consenso entre as 7 segmentações manuais foi obtido através do algoritmo LOP-STAPLE (AKHONDI-ASL et al., 2014). A avaliação dos algoritmos foi realizada em duas categorias: segmentação e detecção de lesões. O desempenho da segmentação foi avaliado com DSC e ASD. Para detecção, o F1-score considera apenas o número de lesões detectadas, não sendo sensível ao volume.

5.1.5 Pré-processamento

O pré-processamento trata de remover ou contornar obstáculos à classificação de imagens. A Figura 7 mostra um exemplo da sequência de tarefas de pré-processamento.

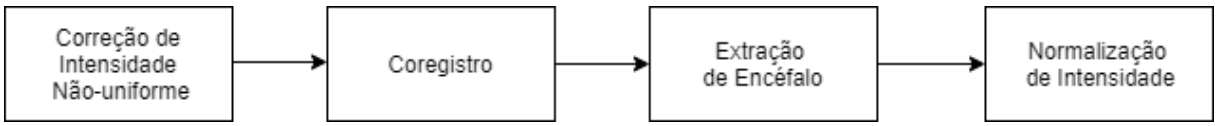

Figura 7 – Estágios de preprocessamento na segmentação de lesões de EM

5.1.5.1 Correção de Intensidade Não-uniforme

Na aquisição das imagens de RM, um artefato comum é uma variação suave nas intensidades observadas, o que pode ser atribuído à baixa uniformidade do campo de radiofrequência, às correntes parasitas e à anatomia do paciente (SLED; ZIJDENBOS; EVANS, 1998). O viés de intensidade não-uniforme pode degradar o desempenho dos métodos de segmentação automática. A Figura 8 mostra um exemplo de correção de intensidade não-uniforme.

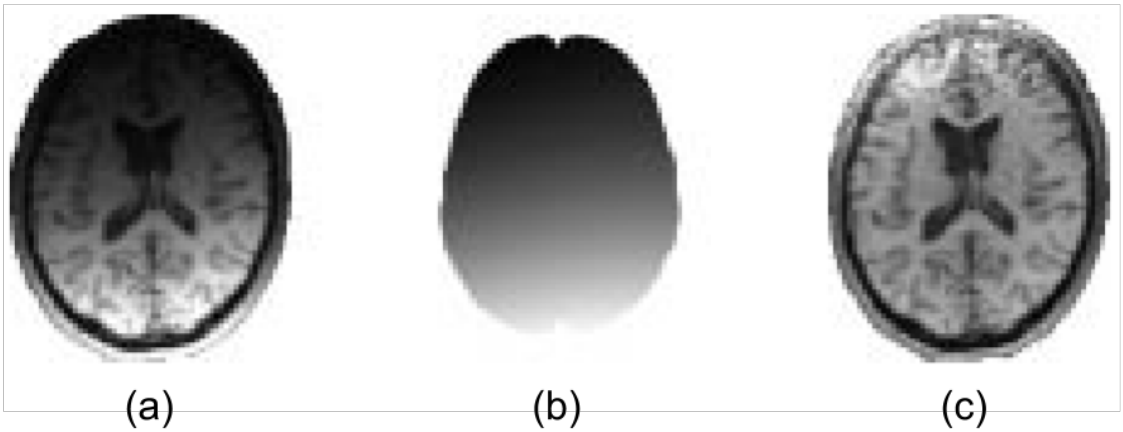

Figura 8 – Correção de intensidade não-uniforme: (a) imagem com artefato de intensidade não-uniforme (b) viés estimado (c) imagem corrigida (adapted from (SLED; ZIJDENBOS; EVANS, 1998)).

Os algoritmos N3 (SLED; ZIJDENBOS; EVANS, 1998) e N4ITK (TUSTISON et al., 2010) estão entre os mais utilizados para realizar correção de não uniformidade de intensidade (BERNAL et al., 2019). Neste trabalho, optamos pela ferramenta N3 para correção de intensidade não-uniforme, por ser amplamente empregada.

5.1.5.2 Corregistro

Nas imagens de RM, os planos axiais adquiridos podem variar em direção, posição relativa e escala entre as sessões de aquisição (COLLINS et al., 1994). Isso resulta em desalinhamento nas imagens de diferentes exames do mesmo paciente. O corregistro transforma imagens de RM para um espaço de coordenadas comum (Figura 9).

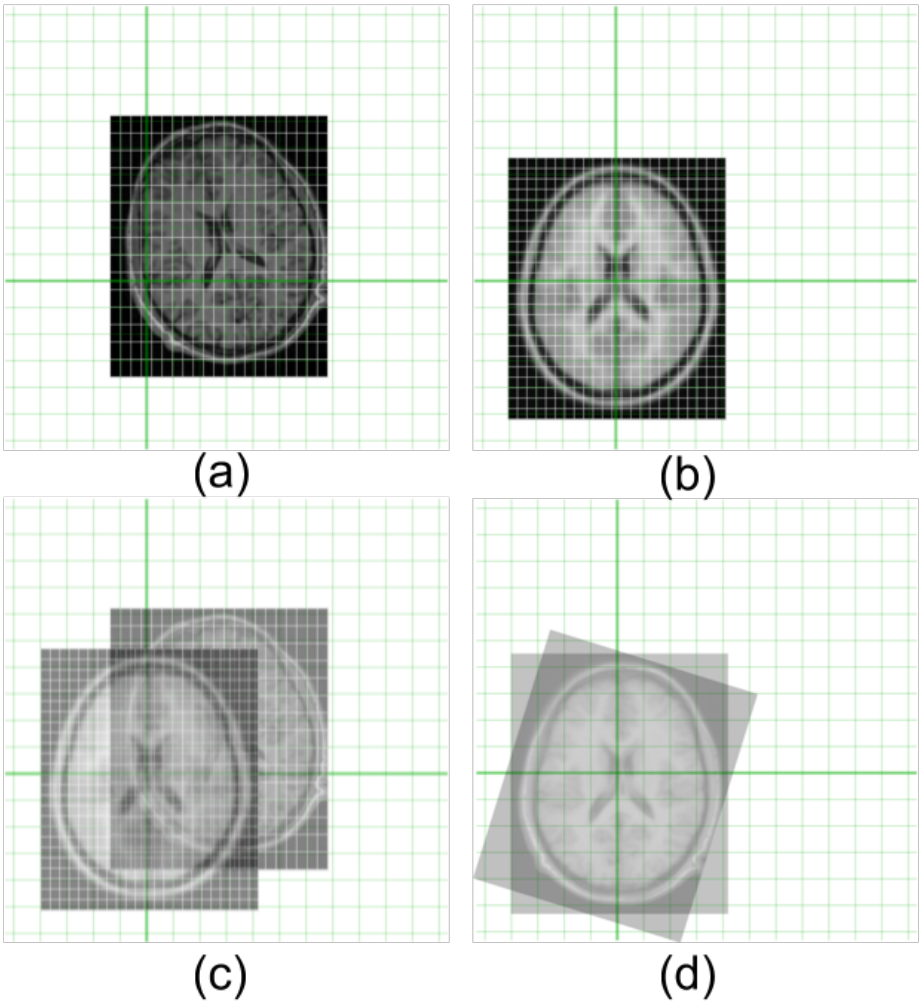

Figura 9 – Corregistro: (a) e (b) são duas imagens de RM do mesmo paciente com diferentes tamanhos de voxel (linhas brancas), posição relativa (linhas verdes) e direção axial; (c) imagens com a mesma posição relativa; (d) imagens registradas (adaptado de (COLLINS; WARD, 2017)).

Neste trabalho, aplicamos a ferramenta mni_autoreg (COLLINS et al., 1994) para corregistro intra-sujeito das demais sequências para o espaço da sequência FLAIR e, também,

para corregistro inter-sujeito de todas as sequências no espaço da sequência FLAIR para o espaço padrão MNI152 (MAZZIOTTA et al., 2001).

5.1.5.3 Extração de Cérebro

Em imagens de ressonância magnética de cabeça, o interesse é frequentemente o tecido cerebral. Além disso, regiões não-cerebrais como olho, medula espinhal, gordura e crânio podem ter um padrão de intensidade semelhante ao de estruturas cerebrais, o que dificulta a segmentação destas estruturas. Assim, o método de extração de cérebro (Figura 10) é aplicado para remover tecidos não-cerebrais antes da classificação (SMITH, 2002; BERNAL et al., 2019). Neste trabalho, realizamos a extração cerebral via ferramenta bet2 (JENKINSON; PECHAUD; SMITH, 2005).

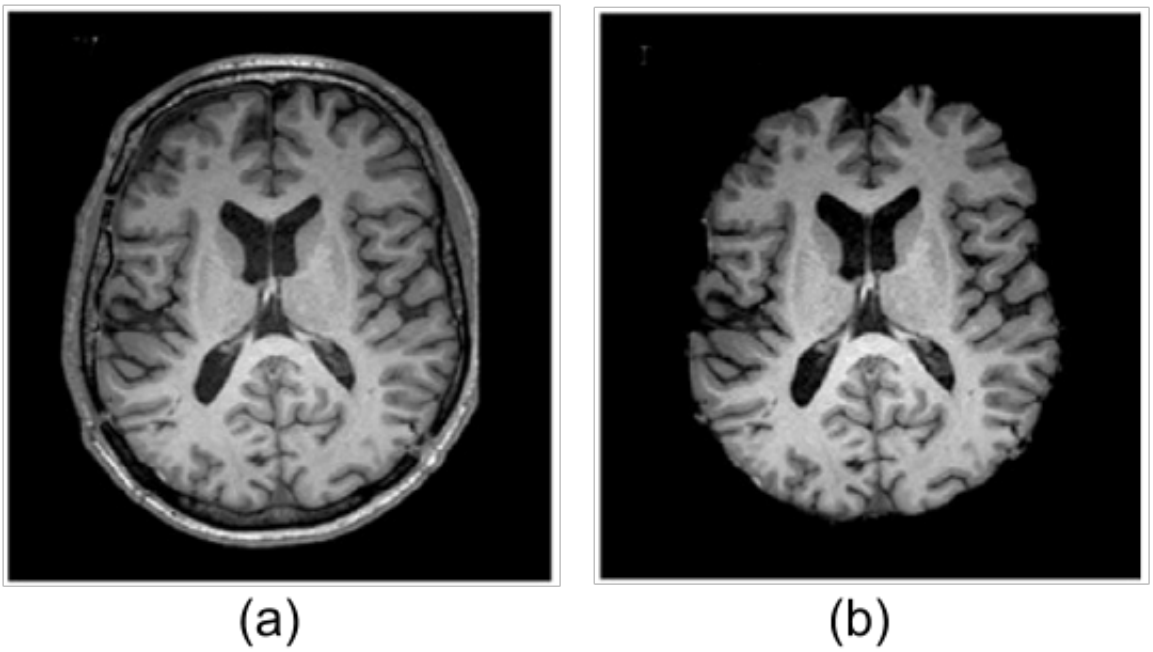

Figura 10 – Extração de cérebro em imagem de ressonância magnética de cabeça: (a) imagem com tecidos não-cerebrais (principalmente, crânio e gordura); (b) imagem do cérebro extraído (adaptado de (BERNAL et al., 2019)).

5.1.5.4 Normalização de Intensidade

Nas imagens de RM, o intervalo de intensidades de *pixel* varia entre sessões de aquisição. A diferença nas distribuições de intensidade é mais evidente ao comparar diferentes equipamentos de RM (SHAH et al., 2011), vide Figura 11, item (a). Entre as muitas abordagem existentes, um método simples e rápido de normalização é a transformação linear das intensidades de cada imagem para um intervalo pré-definido. O objetivo da normalização

é tornar os dados equiparáveis facilitando a segmentação das lesões. Vide Figura 11, no item (b), as intensidades de lesão (ciano) concentradas na mesma região. Neste trabalho, as imagens foram normalizadas para ter mediana 0 e intervalo interquartil 1.

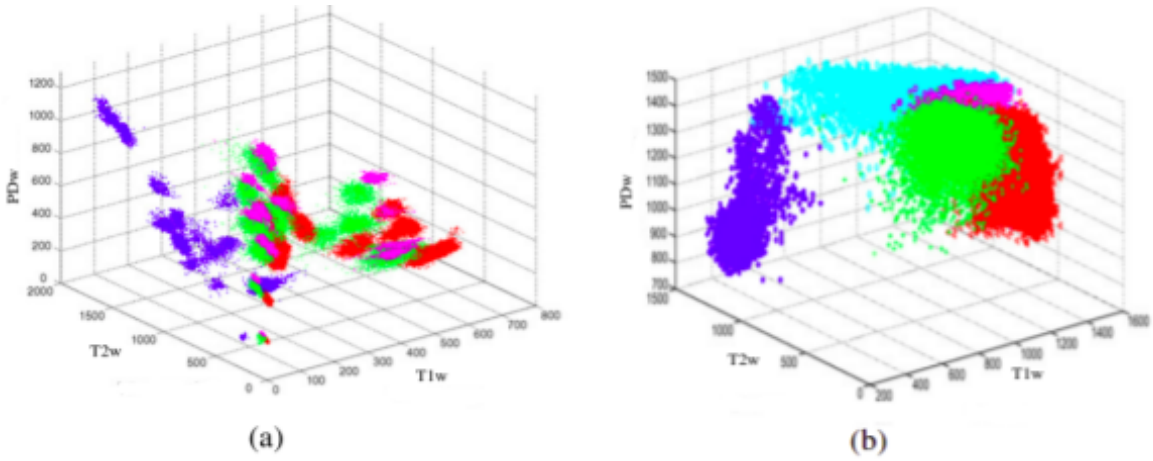

Figura 11 – Normalização de intensidades: intensidades não-normalizadas em (a); intensidades normalizadas em (b). As cores indicam os tecidos: substância branca (em vermelho); substância cinzenta cortical (em verde); substância cinzenta profunda (em magenta); liquor cefalorraquidiano (em azul); lesão (em ciano). A intensidade de lesão é difusa demais para ser exibida em (a). Imagens T1, T2 e PD, adquiridas em 10 equipamentos, de 3 aparelhos de fabricantes diferentes, para 21 pacientes. Adaptado de (SHAH et al., 2011)

5.1.6 Amostragem

O estágio de Amostragem define como os dados pré-processados serão empregados para alimentar o classificador. Neste trabalho empregamos amostragem por *pixel* determinando assim um menor número de entradas no MLP e amenizando a carga computacional. Métodos de amostragem como Aumento de Dados (*Data Augmentation*) e a Extração de Recorte são apropriados para CNNs por explorar correlações locais (VALVERDE et al., 2017), mas poderiam elevar demasiadamente a carga computacional com a rede MLP.

5.1.7 Classificação

Nesta seção, primeiro, definimos o classificador neural e o método de treinamento. Em seguida, definimos o *alvo de aprendizagem* como sendo um componente da função objetivo. Então, definimos as funções objetivo específica e inespecífica.

5.1.7.1 Arquitetura e Treinamento do Perceptron Multicamada

Em nossa primeira abordagem, adotamos um modelo simples de perceptron multicamadas com uma única camada oculta e contendo apenas alguns neurônios, que pode ser

treinado rapidamente e possibilita melhor avaliação estatística. Essa abordagem com uma rede neural superficial serve para demonstrar o impacto do treinamento com função objetivo específica e não deve ser desconsiderada. Redes neurais rasas, quando devidamente parametrizadas e treinadas, podem alcançar um desempenho equivalente ao dos modelos de aprendizagem profunda de última geração (BA; CARUANA, 2014). A arquitetura do perceptron multicamada para segmentação de EM é mostrada na Figura 12 .

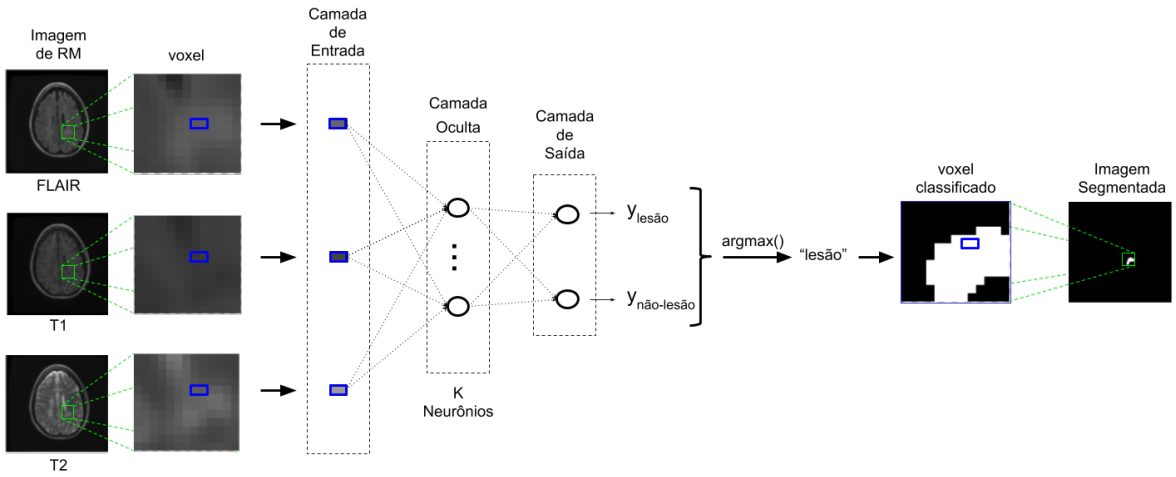

Figura 12 – Arquitetura do MLP para segmentação de lesões de esclerose múltipla em imagens de ressonância magnética. A rede neural recebe três intensidade na camada de entrada, para cada posição da imagem formada por três sequências diferentes de RM (FLAIR, T1 e T2). A camada oculta contém K neurônios totalmente conectados. Na camada de saída, os dois neurônios são também totalmente conectados e representam cada uma das possíveis classes: lesão e não-lesão. O neurônio de saída indica a classe através do valor de ativação máximo, obtido pela função argmax, neste caso indicando corretamente a classe "lesão", produzindo a imagem segmentada

Empregamos a otimização por enxame de partículas (PSO) para treinar o MLP como uma alternativa à retropropagação de erros (EBERHART; SHI, 2007). Assim, usamos o algoritmo MLP-PSO descrito na Seção 2.1.1. O PSO permite maior liberdade na definição da função objetivo sem que seja necessário desenvolver um procedimento específico para cada função a ser otimizada. Em contrate, um algoritmo de retropropagação depende dos gradientes da função objetivo fixada, sendo necessário computar analiticamente os gradientes para cada função objetivo (LECOEUR; FERRÉ; BARILLOT, 2009; BROSCH et al., 2016), o que nem sempre é viável.

5.1.7.2 Alvo de Aprendizagem

O cálculo de métricas de avaliação em 3D seria inviável para treinamento iterativo. Para reduzir a carga computacional, em vez de métricas 3D dispendiosas, empregamos

funções objetivo substitutas mais simples que são computadas em 2D. Assim, a avaliação da segmentação para treinamento depende da média de métricas computadas em 2D. Por exemplo, para o conjunto de dados MICCAI2008, a função objetivo é conforme a seguir: ρ_1: *diferença absoluta relativa da área*; ρ_2: *média da distância simétrica da superfície*; ρ_3: *taxa verdadeira positiva por lesão*; ρ_4: *taxa de falso positivo por lesão*. Como na competição de 2008 (Seção 5.1.3), as quatro métricas são transformadas para ficar no intervalo [0, 100] com uma pontuação de 90 equivalente ao desempenho de um especialista humano. Assim, o alvo de aprendizagem para treinamento orientado a avaliação é

$$\text{2D-Score} = (\rho_1 + \rho_2 + \rho_3 + \rho_4)/4. \tag{5.1}$$

A otimização de todas as quatro métricas é importante: uma única métrica que atinge apenas uma pontuação de 50% representa uma perda de 12,5 pontos na pontuação final, o que é bastante em uma tarefa tão competitiva. A pontuação 2D é mais rápida porque pode ser calculada em uma dimensão menor e calculada em um número reduzido de fatias selecionadas, como abordado mais adiante neste texto. Claramente, o 2D-score resulta em valores diferentes do equivalente tridimensional. Neste trabalho, avaliamos a correlação entre 2D-score e 3D-score como medida de fidelidade das métricas bidimensionais com as métricas tridimensionais. A justificativa para isto é que se duas métricas forem correlacionadas, otimizar uma métrica tenderá a otimizar a outra também.

5.1.7.3 Função Objetivo Específica versus Inespecífica

A função objetivo específica para segmentação de lesões de EM difere da função inespecífica em dois aspectos: alvo de aprendizagem e amostragem de dados. A função objetivo específica (Algoritmo 3) tem como alvo de aprendizagem o 2D-score. Em contraste, a função objetivo inespecífica tem o MSE como um alvo de aprendizagem não específico para segmentação de lesões de EM (Algorithm 4). Em relação à amostragem de dados, a função objetivo específica calcula uma média de quatro métricas 2D que requerem amostras de fatias para cada paciente. Por outro lado, amostras de voxels são usadas para cálculo do MSE. Em comum, a funções objetivo específica e inespecífica incluem uma regra de pré-classificação. Essa regra evita o cálculo desnecessário de saídas do MLP e tira proveito do fato de que as lesões de EM aparecem hiper-intensas nas imagens FLAIR (Algoritmo 3 na linha 5 e Algorithm 4 na linha 3). Observe que essa regra requer intensidades normalizadas. Assim, normalizamos as imagens de modo a ter mediana 0 e intervalo interquartil 1. No treinamento com função inespecífica, para balancear as classes, para cada paciente, todos os voxels das lesões são coletados, e um número igual de voxels não-lesão é amostrado. Observe que a pontuação 2D deve ser maximizada, enquanto a MSE deve ser minimizada. Para cada função objetiva, o PSO foi configurado e empregado de acordo.

Algorithm 3 Função Objetivo Específica

```
1: função objetivoEspecifico(amostra)
2:     para cada paciente em amostra faça
3:         para cada fatia em paciente faça
4:             para cada voxel em fatia faça
5:                 se intensidade FLAIR do voxel < 0 então
6:                     atribua não-lesão ao voxel
7:                 senão
8:                     classifique voxel via MLP
9:                 fim se
10:            fim para
11:            compute 2D-Score (por fatia)    ▷ Section (5.1.7.2)
12:        fim para
13:    fim para
14:    compute 2D-Score médio na amostra
15:    devolve 2D-Score médio
16: fim função
```

Algorithm 4 Função Objetivo Inespecífica

```
1: função objetivoInespecifico(amostra)
2:     para cada voxel em amostra faça
3:         se intensidade FLAIR do voxel < 0 então
4:             atribua não-lesão ao voxel
5:         senão
6:             classifique voxel via MLP
7:         fim se
8:     fim para
9:     compute MSE na amosta
10:    devolve MSE
11: fim função
```

5.1.8 Pós-processamento

Uma das finalidades do pós-processamento é a correção de erros de classificação. Uma etapa de pós-processamento em particular consiste na remoção de regiões consideradas falso positivos. Na segmentação de lesões de EM, a *redução de falso positivos* emprega conhecimento do domínio para descartar lesões candidatas apontadas pelo classificador, aprimorando a segmentação. Em alguns trabalhos, a etapa de redução de falso positivo remove lesões candidatas pequenas, a partir da definição de um limiar para o tamanho da região (BERNAL et al., 2019; VALVERDE et al., 2017; ROURA et al., 2015). Esta abordagem pode ser justificada porque regiões pequenas (com eixo mais longo menor que 3 mm) não são consideradas clinicamente para o diagnóstico da esclerose múltipla (THOMPSON et al., 2018; GRAHL et al., 2019). Nesta tese, cada região tridimensional contígua obtida por componentes conexos e cujo volume fosse menor ou igual a $27mm^3$ foi removida das imagens binárias representando as lesões. A Figura 13 ilustra a remoção de pequenas regiões classificadas como lesão, mas consideras como falso positivos pelo clínico.

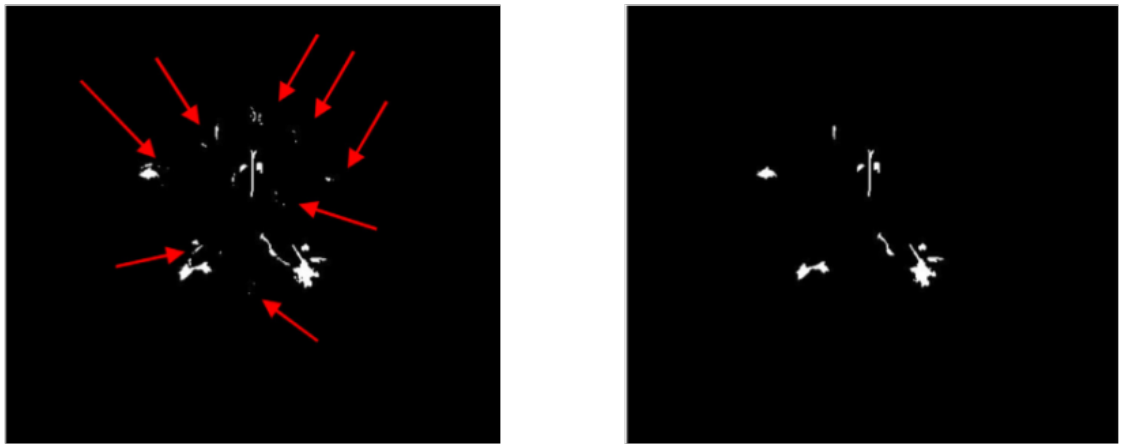

Figura 13 – Redução de falso positivos: (a) imagem segmentada por um algoritmo contendo pequenas regiões de lesão (setas vermelhas); (b) imagem segmentada após a remoção das pequenas regiões consideradas como falso positivos (adaptado de (ALSHAYEJI et al., 2018)).

5.2 MÁQUINA DE APRENDIZADO EXTREMO CONVOLUCIONAL PARA RECONHECIMENTO DE DÍGITOS

Nesta seção, passamos a abordar classificador e tarefa de classificação diferentes da seção anterior, descrevendo a máquina de aprendizado extremo convolucional para reconhecimento de dígitos. Iniciamos apresentando brevemente a base de dados de reconhecimento de dígitos EMNIST, na Seção 5.2.2. Em seguida, apresentamos a arquitetura da CELM proposta e comparamos esta arquitetura com abordagens anteriores, na Seção 5.2.3. Filtros convolucionais, estágios de extração de características e de classificação da CELM são descritos, respectivamente, na Seção 5.2.4, na Seção 5.2.5 e na Seção 5.2.6. Depois, explicamos como aplicar a CELM treinada em dados não vistos, na Seção 5.2.7. Em seguida, propomos uma formulação para o erro de generalização, na Seção 5.2.8.

5.2.1 Ambiente de *Hardware* e *Software*

Os experimentos foram executados em uma máquina virtual na *Google Cloud Platform*[4]. Os serviços de computação em nuvem foram empregados, pois permitem acesso a hardware avançado de maneira econômica. Uma máquina virtual com 64 CPUs virtuais e 256 GB de memória RAM foi alocada. Os algoritmos foram executados no Matlab 2017a, em um sistema operacional Ubuntu 17.04.

[4] Em <https://cloud.google.com>, último acesso em 13/12/2019

5.2.2 Conjunto de Dados EMNIST

O banco de dados EMNIST consiste em imagens de dígitos e de letras manuscritas (COHEN et al., 2017). Cada imagem consiste em pixels de 8 bits em escala de cinza de tamanho 28×28. Por um lado, a partição com classes balanceadas EMNIST-Digits-Balanced contém 240.000 exemplos de dígitos para treinamento e 40.000 exemplos para teste. Por outro lado, para classes não balanceadas, a partição EMNIST-Digits-all contém 344.307 dígitos para treinamento e 58.646 para teste, o que equivale a todos os dígitos no NIST-SD-19.

5.2.3 Arquitetura da CELM

A arquitetura proposta foi construída a partir da CELM rasa de (MCDONNELL; VLADUSICH, 2015). Em trabalho anterior, uma extensão da CELM rasa com adição de um segundo estágio convolucional não produziu resultados tão acurados quanto com apenas um estágio (YOUSEFI-AZAR; MCDONNELL, 2017). Para esse tipo de CELM, mostramos que dois estágios de extração de características podem resultar em um melhor poder discriminativo do que com apenas um estágio. A Figura 14 compara a nossa arquitetura proposta com as redes em (HUANG et al., 2015b) e (MCDONNELL; VLADUSICH, 2015). Nessa figura, as camadas são ilustrativamente representadas por operações de matrizes, as funções de ativação e de solução de mínimos quadrados são denotadas, respectivamente, ϕ and ψ. Além disso, nas camadas de *pooling*, as exponenciações e a raízes quadradas são devido à operação Lp-pooling com parâmetro $P = 2$ (Seção 5.2.5). À esquerda, a LRF-ELM emprega filtros aleatórios para extração de características, possui uma única camada na etapa de classificação e realiza amostragem na camada de *pooling*, não reduzindo o tamanho do mapa de características, com $\mathbf{W}_{pool}$ atuando como indicador da área amostrada. Ao centro, a CELM de McDonnell e Vladusich (2015) emprega combinação de filtros para detecção, realiza subamostragem no estágio de pooling e possui duas camadas no estágio de classificação. À direita, a arquitetura proposta emprega dois estágios de extração de características e usa diferentes tipos de filtros e combinações em relação a rede de McDonnell e Vladusich (2015). Note que a arquitetura da CELM proposta é apresentada nesta seção como um fluxograma por compacidade na comparação com abordagens anteriores. Alternativamente, o fluxo de processamento através das camada de entrada, convolucional, e subamostragem na CELM proposta é semelhante ao fluxo que acontece na CNN da Figura 2, na Seção 1.1, diferindo no tamanho da imagem de entrada que é 28×28 para o conjunto de dados EMNIST.

.

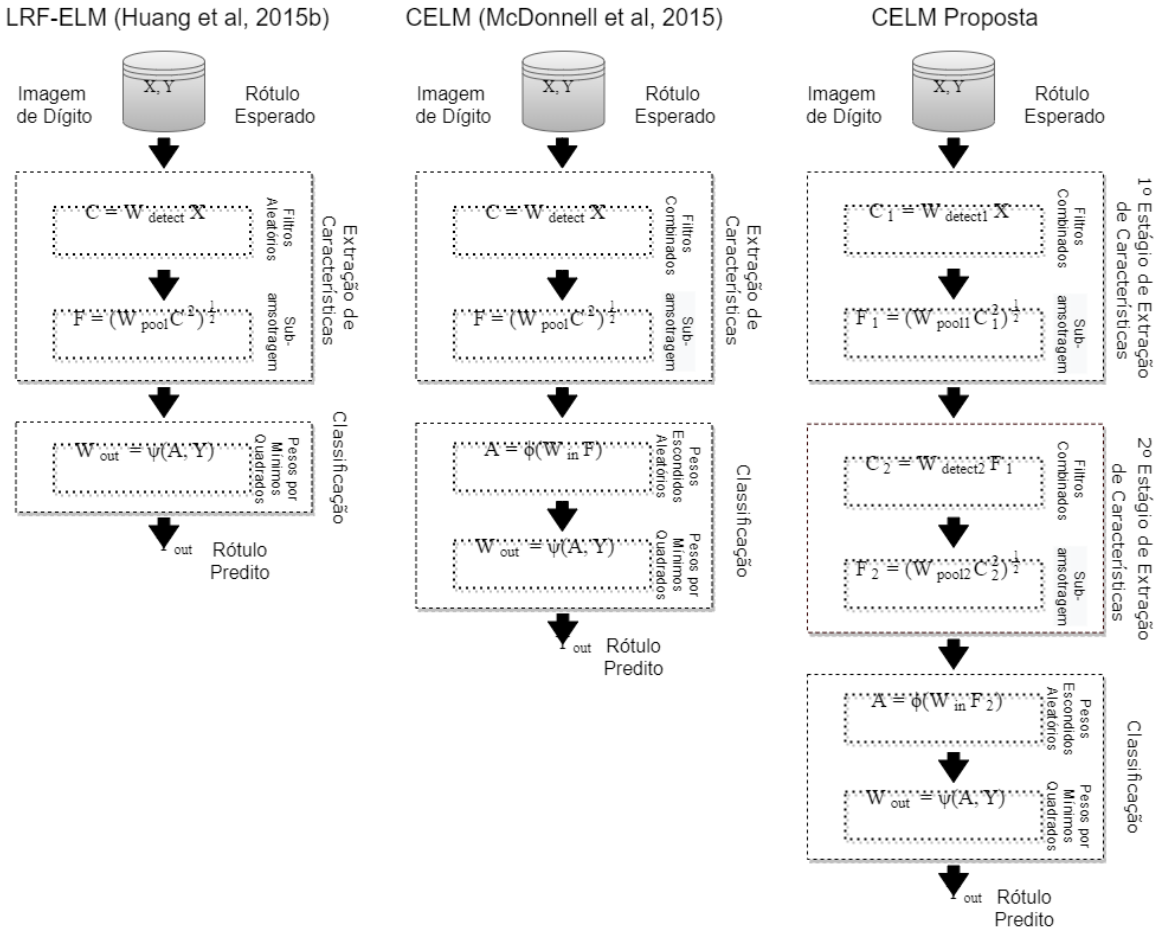

Figura 14 – Comparação da arquitetura da CELM proposta com duas abordagens anteriores. A arquitetura da CELM proposta diferencia-se das anteriores por empregar dois estágio de extração de características e novas combinações de filtros convolucionais (em vermelho), *resultando em maior acurácia.*

5.2.4 Filtros Convolucionais

Nesta seção, focamos em filtros convolucionais atuando como detectores de padrões. A seguir, descrevemos os filtros empregados nos estágios de extração de características da CELM proposta.

- **Filtro Aleatório** tem cada elemento definido por um número pseudo-aleatório com base em uma distribuição de probabilidade (JARRETT et al., 2009; CHAN et al., 2015; YOUSEFI-AZAR; MCDONNELL, 2017). Este tipo de filtro tem sido usado como linha de base para comparação com outros filtros. Entretanto, com certa surpresa, resulta em degradação apenas moderada do poder discriminativo (JARRETT et al., 2009; CHAN et al., 2015).

- **Filtro Recorte** é uma sub-região selecionada a partir de uma imagem. Nós usamos apenas recortes da região central de imagens selecionadas aleatoriamente, seguindo

uma abordagem em (MCDONNELL; VLADUSICH, 2015).

- **Filtro PCA** é formado a partir de um conjunto de recortes de imagens. Os recortes vetorizados são dispostos em uma matriz para extração dos componentes principais (CHAN et al., 2015).

- **Filtro de Gabor** é um modelo matemático que abrange um conjunto de filtros de valores complexos que variam em orientação, frequência e dimensão espacial (DAUGMAN, 1985). No campo receptivo de neurônios sensoriais chamados células simples, o padrão espacial de resposta a um estímulo luminoso corresponde bem a filtros de Gabor no domínio espacial (DAUGMAN, 1985; JONES; PALMER, 1987). Deste modo, filtros de Gabor representam detectores de padrões que ocorrem na natureza.

5.2.5 Estágio de Extração de Características

Apresentamos uma breve descrição do estágio convolutivo, mais detalhes podem ser encontrados em (MCDONNELL; VLADUSICH, 2015). As camadas de detecção e de *pooling* compõe cada estágio de extração de características e são descritas a seguir.

Em uma camada de detecção, as convoluções são realizadas usando multiplicação de matrizes, o que permite rápida extração de características. Assim, para cada imagem $\mathcal{I}_i$ de tamanho $J \times J$ *pixels*, a matriz $\mathbf{X} \in \mathbb{R}^{J^2 \times M}$ contém M imagens vetorizadas $\mathbf{x}_i = \text{vec}(\mathcal{I}_i)$. Além disso, cada um de F filtros de tamanho $W \times W$ é convertido em uma matriz de Toeplitz denotada $\mathbf{W}_{detect} \in \mathbb{R}^{L^2 \times J^2}$, com $L = (J+W-1)$ para um mapa de características de tamanho $L \times L$. A multiplicação $\mathbf{W}_{detect}\mathbf{X}$ realiza a convolução de todas as imagens na matriz de entrada para cada filtro.

A camada de *pooling* também é baseada em convolução por multiplicação de matrizes, mas com o propósito de redução do espaço de características. Essa camada realiza filtragem passa-baixa e subamostragem em uma única convolução. Para filtragem passa-baixa, um filtro de suavização espacial de tamanho $Q \times Q$ é usado para formar uma matriz de Toeplitz $\mathbf{P} \in \mathbb{R}^{V^2 \times L^2}$ com $V = (J + Q - 2W + 1)$ sendo o tamanho da região válida da convolução. Para subamostragem, a matriz $\mathbf{P}$ é amostrada em pontos que são determinados pelo tamanho do passo D, reduzindo a dimensão espacial do mapa de características em $U = \lceil V/D \rceil$ e resultando na matriz $\mathbf{W}_{pool} \in \mathbb{R}^{U^2 \times J^2}$. Após aplicação das camadas de detecção e *pooling*, a matriz com os mapas de características para um único filtro $\mathbf{F}_{single}$ é dado por:

$$\mathbf{F}_{single} = (\mathbf{W}_{pool}(\mathbf{W}_{detect}\mathbf{X})^p)^{\frac{1}{p}}. \tag{5.2}$$

Nesta equação, as exponenciações são realizadas por elemento da matriz, definindo a operação conhecida como Lp-pooling com parâmetro P (SERMANET; CHINTALA; LECUN, 2012). Em comparação, McDonnell e Vladusich (2015) empregou também pooling com

subamostragem, mas substituindo o filtro Gaussiano passa-baixa usado em (SERMANET; CHINTALA; LECUN, 2012) por um filtro de média. Qual filtro de pooling é mais adequado pode depender dos dados em mãos. Em nosso trabalho, selecionamos o filtro Gaussiano no pooling o que resultou em acurácia ligeiramente melhor que o filtro de média.

Depois da aplicação de F filtros, os mapas de características resultantes são agregados, com $G = FU^2$ características para cada um dos M exemplos. Assim, a matriz de mapas é tal que $\mathbf{F}_{all} \in \mathbb{R}^{G \times M}$

Ao final de um estágio convolutivo, a matriz de mapa de característica é normalizada como segue:

$$\mathbf{F} \equiv \left(\frac{\mathbf{F}_{all}}{F_{max}} \right)^{\frac{1}{2}}, \tag{5.3}$$

com $F_{max} = \max \mathbf{F}_{all}$ e todas operações realizadas por elemento.

Estágios de extração de características podem ser cascateados, aumentando a profundidade da rede e, potencialmente, o poder discriminativo. Deste modo, um mapa de características na saída de um estágio é tratado como entrada de outro estágio de extração de características, seguindo a abordagem em (CHAN et al., 2015).

5.2.6 Estágio de Classificação

O estágio de classificação mapeia as características extraídas aos rótulos de classe, independentemente do número de estágios de extração de características encadeados. O estágio de classificação é dividido em duas operações: camada oculta e camada de saída.

O primeiro passo na computação da camada oculta é a geração de pesos aleatórios. No estágio de classificação, do mesmo modo como ocorre com os filtros convolucionais na extração de características, números aleatórios simples são frequentemente usados para gerar pesos ocultos em redes neurais de treinamento rápido (HUANG et al., 2015a; RAHIMI; RECHT, 2009). Em vez de pesos aleatórios simples, nós seguimos uma abordagem que gera aleatoriamente pesos ocultos a partir de um conjunto restrito de vetores de diferenças de amostras entre classes, obtendo melhor poder discriminativo (ZHU; MIAO; QING, 2014).

Para geração de pesos ocultos, sejam $\mathbf{f}_{class1}$ e $\mathbf{f}_{class2}$ vetores de características tomados ao acaso das colunas da matriz de mapa de característica $\mathbf{F}$ para as classes *classe1* e *classe2*, respectivamente, então o vetor de diferença é definido por

$$\mathbf{d} = \frac{2(\mathbf{f}_{classe1} - \mathbf{f}_{classe2})}{\|\mathbf{f}_{classe1} - \mathbf{f}_{classe2}\|^2}, \tag{5.4}$$

em que $\|.\|$ é a norma euclidiana. O *bias* b é dado por

$$b = \frac{(\mathbf{f}_{class1} + \mathbf{f}_{class2})^T (\mathbf{f}_{class1} - \mathbf{f}_{class2})}{\|\mathbf{f}_{class1} - \mathbf{f}_{class2}\|^2}. \tag{5.5}$$

Um vetor de pesos é então gerado por meio da seguinte concatenação $\mathbf{w} = [\mathbf{d}', b] \in \mathbb{R}^{G+1}$, em que $(.)'$ denota transposta. A geração é repetida para K vetores de pesos.

Os vetores gerados são usados para forma a matriz de pesos ocultos tal que $\mathbf{W}_{hidden} \in \mathbb{R}^{K \times G+1}$. Além disso, um vetor contendo M uns $\mathbf{1} = [1, \ldots, 1]$ é concatenado após a última linha da matriz de características tal que $\mathbf{F} \in \mathbb{R}^{G+1 \times M}$, devido ao *bias* concatenado previamente.

Por fim, a função de ativação $\phi(.)$ é aplicada por elemento ao resultado da multiplicação das matrizes de pesos ocultos e de mapas de características, isto é

$$\mathbf{A} = \phi(\mathbf{W}_{hidden}\mathbf{F}), \tag{5.6}$$

em que $\mathbf{A} \in \mathbb{R}^{K \times M}$.

Para cálculo da camada de saída, os pesos são calculados como na máquina de aprendizado extremo para classificação multiclasses com múltiplas saídas (HUANG et al., 2012). Deste modo, os rótulos de P classes são representados na matriz $\mathbf{Y}$, tal que cada elemento do vetor coluna $\mathbf{y}_j = [y_{1,j}, \ldots, y_{i,j}, \ldots, y_{P,j}]'$ contém 1 se o exemplo j é da classe i e 0 caso contrário. Dada a matriz de características da camada escondida $\mathbf{A}$ e a matriz de rótulos $\mathbf{Y}$, o problema de treinamento do classificador envolve encontra $\mathbf{W}_{out} \in \mathbf{R}^{P \times K}$ tal que $\mathbf{Y} \approx \mathbf{W}_{out}\mathbf{A}$. Huang et al. (2012) fornecem duas alternativas para cálculo dos pesos de saída dependendo do tamanho do conjunto de treinamento (M) e do número de neurônios ocultos (K). Se há mais exemplos que neurônios ocultos ($M > K$), a matriz de pesos de saída é dada por

$$\mathbf{W}_{out} = \mathbf{Y}\mathbf{A}'(\mathbf{A}\mathbf{A}' + C\mathbf{I})^{\dagger}, \tag{5.7}$$

em que $\mathbf{I}$ é uma matriz identidade $K \times K$, C é um hiper-parâmetro e $(.)^{\dagger}$ denota a inversa generalizada de Moore-Penrose .

Caso o número de neurônios oculto seja maior que o número de exemplos de treinamento ($M < K$), os pesos de saída são

$$\mathbf{W}_{out} = \mathbf{Y}(\mathbf{A}'\mathbf{A} + C\mathbf{I})^{\dagger}\mathbf{A}'. \tag{5.8}$$

Uma vez que o classificador está treinado, a arquitetura como um todo pode ser aplicada a novos dados.

É importante notar que qualquer uma das duas alternativas pode ser aplicada independentemente das condições apresentadas. No entanto, as alternativas têm custos computacionais diferentes para os quais o número de neurônios escondidos e o número de exemplos de treinamento devem ser observados. O cálculo dos pesos de saída encerra o treinamento do estágio de classificação.

5.2.7 Aplicação em Dados Novos

Para aplicar a CELM treinada a um conjunto de dados não vistos, submetemos as imagens à entrada da rede, empregando nos estágios de extração e classificação os filtros

$\mathbf{W}_{detect}$ e $\mathbf{W}_{pool}$ (Seções 5.2.4 e 5.2.5), os pesos $\mathbf{W}_{hidden}$ e $\mathbf{W}_{out}$ (Seção 5.2.6) e demais parâmetros, previamente definidos. Dada a matriz $\mathbf{X}_{test}$ com imagens vetorizadas do conjunto de teste. Aplicando as seguintes equações

$$\mathbf{F}_{single\_test} = (\mathbf{W}_{pool}(\mathbf{W}_{detect}\mathbf{X}_{test})^p)^{\frac{1}{p}} \tag{5.9}$$

e

$$\mathbf{F}_{test} = \left(\frac{\mathbf{F}_{all\_test}}{F_{max}}\right)^{\frac{1}{2}}, \tag{5.10}$$

agregamos $\mathbf{F}_{single\_test}$ com F filtros em $\mathbf{F}_{all\_test}$, temos a matriz de rótulos aproximada

$$\mathbf{Y}_{test} = \mathbf{W}_{out}\mathbf{A}_{test} = \mathbf{W}_{out}\phi(\mathbf{W}_{hidden}\mathbf{F}_{test}). \tag{5.11}$$

Por fim, para cada vetor coluna $\mathbf{y}_j$ da matriz de rótulos aproximada $\mathbf{Y}_{test}$, o rótulo de saída é dado por

$$\text{label}(\mathbf{y}_j) = \underset{i\in\{1,\ldots,P\}}{\text{argmax}}\, y_{i,j}. \tag{5.12}$$

5.2.8 Modelo e Predição do Erro de Generalização

Nossa formulação para o erro de generalização é inspirada no limite superior dado por um teorema em (RAHIMI; RECHT, 2009). A fim de expor esse resultado teórico, considere o problema de treinar um modelo $f : \mathcal{X} \to \mathbb{R}$ a partir de um conjunto de pares $\{\mathbf{z}_i, y_i\}_{i=1}^M$ de exemplos $\mathbf{z}_i \in \mathcal{X}$ e rótulos de classe $y_i \in \{-1, 1\}$ que são obtidos *iid* de uma distribuição de probabilidade $P(\mathbf{z}, y)$. Defina uma função de custo $c(f(\mathbf{z}), y)$ que penaliza o desvio entre o rótulo estimado pelo modelo $f(\mathbf{z})$ e rótulo alvo y. O Teorema a seguir fornece um limite superior para o erro de generalização.

Teorema 1 *(Rahimi-Recht) Para parâmetros $\mathbf{w}_1, \ldots, \mathbf{w}_K$ (vetores de pesos ocultos) amostras de uma distribuição p arbitrária sobre Ω, a função de ativação ϕ tal que $|\phi(\mathbf{z}; \mathbf{w})| \leq 1$ e parâmetros $\boldsymbol{\psi} = \{\psi_1, \ldots, \psi_K\}$ (vetor de pesos de saída), defina o conjunto de funções $\mathcal{F} = \{f(\mathbf{z}) = \int_\Omega \psi(\mathbf{w})\phi(\mathbf{z}; \mathbf{w})d\mathbf{w} \mid |\phi(\mathbf{z}; \mathbf{w})| \leq Cp(\mathbf{w})\}$. Além disso, sejam c uma função de custo L-Lipschitz e $\delta > 0$. Defina o risco empírico $R_{K,M}[\hat{f}] = \frac{1}{M}\sum_{i=1}^M c(\hat{f}(\mathbf{z}_i), y_i)$ com $\hat{f}(\mathbf{z}) = \sum_{j=1}^K \psi_j\phi(\mathbf{z}; \mathbf{w}_j)$ minimizado pelo vetor $\boldsymbol{\psi}$. Defina também o risco verdadeiro (desconhecido) $R[f] = \mathbb{E}_P\, c(f(\mathbf{z}), y)$ sobre dados inéditos. Então a diferença entre risco empírico e mínimo risco verdadeiro é limita por*

$$R_{K,M}[\hat{f}] - \min_{f\in\mathcal{F}} R[f] \leq \mathcal{O}\left(\left(\frac{1}{\sqrt{K}} + \frac{1}{\sqrt{M}}\right) LC\sqrt{\log\frac{1}{\delta}}\right) \tag{5.13}$$

com probabilidade ao menos $1 - 2\delta$.

Na aplicação desse resultado teórico a CELM proposta, ressaltamos duas condições. Primeiro, o teorema supõe que a função de ativação é limitada com $|\phi| \leq 1$, o que restringe

um pouco o tipo de função de ativação. No entanto, a normalização de característica (Equação 5.3) faz com que o domínio de ϕ seja $[-1, 1]$. Deste modo, a função $\phi(x) = x^2$ satisfaz a condição de limitação com contradomínio limitado em $[-1, 1]$. Segundo, o teorema assume rótulos de classe univariados. Nós conjecturamos que o limite superior do erro permanece para mais de duas classes. De fato, nossos experimentos suportam essa alegação.

Baseado no Teorema 1, um modelo para o erro médio de teste é aproximado por:

$$e_{K,M}(\hat{f}) \equiv \frac{C_K}{\sqrt{K}} + \frac{C_M}{\sqrt{M}}. \tag{5.14}$$

Assim, o erro médio de teste $e_{K,M}$, para uma rede treinada $\hat{f}$, é proporcional a soma de dois termos que decaem com aumento de neurônios na camada oculta K e aumento de exemplos no conjunto de treinamento M. As constantes C_K e C_M são fatores de escala em cada termo. A fim de predizer o erro, calculamos erros para valores moderados de K e M por validação cruzada. Em seguida, realizamos um ajuste de curva para encontrar C_K e C_M. Então, usamos o estimador ajustado para predizer o erro para valores maiores e mais custosos dos argumentos. Até onde sabemos, está é a primeira validação empírica do modelo de Rahimi-Reicht envolvendo ao mesmo tempo o tamanho do conjunto de treinamento M e o número de neurônios escondidos K.

6 EXPERIMENTOS E RESULTADOS

Este capítulo está dividido em quatro seções. Cada seção descreve experimentos e resultados para cada abordagem proposta em um conjunto de dados específico. As Seções 6.1 e 6.2 tratam da aplicação de funções objetivos específicas para segmentação de lesões de EM, respectivamente, nos conjuntos de dados MICCAI2008 e MICCAI2016. As máquinas de aprendizado extremo convolucionais são abordadas na Seção 6.2.2.1, no reconhecimento de dígitos.

6.1 FUNÇÃO OBJETIVO PARA SEGMENTAÇÃO DE ESCLEROSE MÚLTIPLA NA BASE DE DADOS MICCAI2008

Esta seção está organizada como segue. A configuração experimental inclui o modo de avaliação do alvo de aprendizagem usado na função objetivo específica (Seção 6.1.1.1), além do uso do conjunto de dados MICCAI2008 e treinamento do classificador (Seção 6.1.1.2). A apresentação dos resultados está assim dividida: avaliação da fidelidade e eficiência do alvo de aprendizagem, na Seção 6.1.2.1; comparação da eficiência de treinamento entre diferentes funções objetivo, na Seção 6.1.2.2; comparação do desempenho de segmentação no MICCAI2008 entre diferentes funções objetivo, na Seção 6.1.2.3, e entre diferentes algoritmos do Estado-da-Arte, na Seção 6.1.2.4.

6.1.1 Configuração Experimental

6.1.1.1 Fidelidade e Eficiência do Alvo de Aprendizagem

A função objetivo com o 2D-score (vide Equação 5.1) foi avaliada quanto à fidelidade e eficiência em função do número de fatias de imagens de RM. O plano, a quantidade e a posição das fatias selecionadas podem influenciar o 2D-Score médio. Fatias no plano axial foram selecionadas, que variaram em número s da seguinte forma: $s = \{2^2, 2^3, \ldots, 2^9\}$. A posição p_n da n-ésima fatia foi dada por $p_n = 1 + (n-1)(\frac{2^9}{s})$ quando s fatias foram selecionadas. Por exemplo, para $s = 4$, as fatias selecionadas foram $\{1, 129, 257, 385\}$. Além disso, avaliamos um caso com três fatias mediais nas posições 240, 250 e 260. Um 2D-Score baseado em s fatias por paciente é denotado como 2D-s.

Para esta análise, foi empregado um subconjunto de 10 imagens segmentadas por 2 especialistas humanos, totalizando 20 segmentações. Para cada uma dessas segmentações, o 2D-Score e seu análogo tridimensional (3D-Score) foram computados como medidas de concordância avaliador-avaliador. A fidelidade foi dada pela correlação de Pearson do 2D-Score com o 3D-Score. A eficiência foi avaliada através do tempo mediano por paciente para calcular cada função objetivo.

6.1.1.2 Uso do Conjunto de Dados e Treinamento do Classificador

Usando as imagens do conjunto de treinamento, empregamos o método de validação-cruzada *holdout* repetido para comparar os algoritmos estatisticamente. Para cada tipo de treinamento, a execução do MLP-PSO foi repetida 120 vezes. Nós comparamos duas funções objetivo para treinamento: específica para segmentação de EM, com o 2D-score-3 (Seção 6.1.1.1); inespecífica, com o MSE. Calculamos o tempo médio de execução e o número médio de avaliações da função objetivo para avaliar a eficiência de treinamento.

Comparamos as funções objetivo específica e inespecífica usando também as imagens do conjunto de teste. Assim, para cada função objetivo, selecionamos o MLP com o melhor desempenho de segmentação no conjunto de treinamento. Em seguida, os MLP selecionados para cada função objetivo foram aplicados ao conjunto de imagens de teste. Então, as funções objetivo foram avaliadas através do 3D-score fornecido pelo site da competição MICCAI2008.

6.1.2 Resultados

6.1.2.1 Avaliação da Fidelidade e Eficiência do Alvo de Aprendizagem

O 2D-score, componente da função objetivo proposta, é comparado ao 3D-score, a métrica final de segmentação da competição de segmentação MICCAI2008. As métricas são comparadas quanto à fidelidade e à eficiência. A fidelidade tende a crescer com o número de fatias usadas para calcular a pontuação 2D (Tabela 4). Por outro lado, a eficiência do 2D-score diminui com mais fatias. Observe que o 2D-score tem fidelidade positiva, mesmo com um pequeno número de fatias. Uma correlação positiva, mesmo fraca, indica uma tendência para um maior 3D-score, consequentemente melhor segmentação, à medida que o 2D-score aumenta. Esta relação monotônica torna razoável o uso do 2D-score-3 de baixa fidelidade na função objetivo específica. Além disso, o 2D-score-3 de baixa fidelidade é cerca de 80 vezes mais rápido que o 2D-score-512 de alta fidelidade.

Tabela 4 – Fidelidade e Eficiência do 2D-score em Relação ao 3D-score

	Número de Fatias no 2D-score								
	3	4	8	16	32	64	128	256	512
Correlação com 3D-score	0,13	0,15	0,17	0,19	0,28	0,29	0,49	0,77	0,81
Tempo Mediano por Paciente (s)	0,04	0,04	0,06	0,11	0,22	0,44	0,88	1,73	3,46

6.1.2.2 Eficiência de Treinamento e Função Objetivo

Comparamos as funções objetivo específica e inespecífica (5.1.7.3) com relação ao número de avaliações e tempo total de treinamento do MLP-PSO (Tabela 5). O treinamento com função objetivo específica requereu um menor número de avaliações, mas levou mais tempo em geral ($\sim$ 1 hora) em comparação com treinamento com função objetivo inespecífica ($\sim$ 1 minuto), porque o MSE é computado mais rápido que as métricas 2D. No treinamento com função objetivo específica, o tempo por avaliação da função objetivo é de aproximadamente $3238/1645 = 1,968s$. O treinamento com função objetivo inespecífica é mais eficiente levando 0,027 segundo por avaliação. No entanto, a função objetivo inespecífica resulta em um baixo desempenho de segmentação, como será visto adiante.

Tabela 5 – Tempo (s) de Treinamento por Função Objetivo

	Inespecífica	Específica
Número de Avaliações	2911	1645
Tempo de treinamento	78,81	3238,48
Tempo por avaliação	0,027	1,968

A função objetivo específica poderia empregar diretamente o 3D-score, ao invés do 2D-score, mas isto seria pouco eficiente, como destacado a seguir. Com o 2D-score e 10 indivíduos no conjunto de treinamento, o tempo por paciente em um único cálculo da função objetivo específica é de $0,1968s$ (note que o tempo para obter apenas o 2D-score-3 é de $0,04s$ por paciente). Assim, o tempo total de treinamento com o 2D-score é aproximadamente 1645 (avaliações) $\times$ 10 (pacientes) $\times$ 0,1968 (segundo por avaliação) = $\sim$ 1 hora. Por outro lado, computar o 3D-score para uma única segmentação leva cerca de 120 segundos. Logo, com o 3D-score na função objetivo específica, o tempo total de treinamento seria 1645 (avaliações) $\times$ 10 (pacientes) $\times$ 120 (segundos por avaliação) = $\sim$ 3 semanas.

6.1.2.3 Comparação de Funções Objetivo para Segmentação no MICCAI2008

A função objetivo específica resultou em melhor desempenho de segmentação em geral (3D-score = 83,26%) comparado ao treinamento com função objetivo inespecífica (3D-score=56,30%), no conjunto de imagens de treinamento do MICCAI2008. Isto pode ser visto na Tabela 6, em que as melhores pontuações estão em negrito. Note que as métricas estão transformadas em pontuação entre 0 e 100, quanto maior, melhor (Vide Seção 5.1.3). A função objetivo inespecífica teve melhor desempenho apenas em verdadeiro positivo por lesão (LTPR) entre as métricas que compõem o 3D-score, mas o mal desempenho em falso positivo por lesão (LFPR) pode indicar sobressegmentação (vide Seção 2.3.2.4). Estes resultados revelam em quais métricas a função objetivo específica leva vantagem.

Além disso, a função objetivo específica proposta obteve um melhor balanço das quatro métricas componentes do 3D-score, superando a função objetivo inespecífica na pontuação média. Assim, o 2D-score-3 com apenas três fatias, apesar da baixa fidelidade em relação ao 3D-score, permitiu atingir uma bom desempenho de segmentação.

A simplicidade do perceptron multicamada pode explicar o baixo desempenho de segmentação de EM com a função objetivo inespecífica, o que seria diferente se uma rede neural profunda fosse usada mesmo com uma função objetivo inespecífica. Treinada com função objetivo específica, uma rede neural profunda do estado-da-arte poderia resultar em desempenho de segmentação de EM ainda melhor do que o perceptron multicamada. No entanto, a otimização de uma rede neural profunda usando a função objetivo específica seria dificultada pela ausência de algoritmo de retropropagação específico (como mencionado na Seção 5.1.7.1) e pela inviabilidade do PSO em um espaço de busca de alta dimensão determinado pelos milhares de pesos ajustáveis desta rede (CHEN; MONTGOMERY; BOLUFÉ-RÖHLER, 2015).

Tabela 6 – Métricas de Segmentação de EM por Função Objetivo no MICCAI2008

Função Objetivo	RAVD	ASSD	LTPR	LFPR	Média (3D-Score)
Inespecífica	6	71	**98**	51	56,30
Específica	**91**	**87**	76	**80**	**83,26**

6.1.2.4 Comparação de Algoritmos de Segmentação no MICCAI2008

Comparado com abordagens anteriores, o método proposto é o único a otimizar diretamente as métricas de segmentação LTPR, LFPR, RAVD e ASSD que são também usadas para avaliação no MICCAI2008 (Tabela 7). Além disso, usando a função objetivo específica, obtivemos um desempenho de segmentação comparável aos melhores algoritmos no MICCAI2008. Curiosamente, nosso método baseado em uma rede MLP simples e rasa atingiu um desempenho de segmentação com 3D-score de 83,26 que é muito próximo do que foi obtido por uma rede neural profunda com 84,07 (BROSCH et al., 2016). Note que o método com duas CNNs em cascata de Valverde et al. (2017) avançou o estado-da-arte sobre o conjunto de dados MICCAI2008 com 3D-score=87,11 posteriormente à publicação dos resultados com o nosso método baseado em MLP (SANTOS et al., 2016b).

Tabela 7 – Comparação de Algoritmos de Segmentação de EM no MICCAI2008

Função Objetivo	3D-Score	Referência
Cross-entropy	**87,12**	(VALVERDE et al., 2017)
Maximum a Posteriori	86,93	(JESSON; ARBEL, 2015)
Rotation-invariant Distance	86,10	(GUIZARD et al., 2015)
Trimmed Likelihood, *Energy Function*	84,46	(TOMAS-FERNANDEZ; WARFIELD, 2015)
MSE *Weighted by Sensitivity*	84,07	(BROSCH et al., 2016)
⇒ LTPR, LFPR, RAVD, ASSD	83,26	presente trabalho
Energy Function, ROC, Jaccard	82,54	(ZHAN et al., 2015)
Dice (*voxel-wise*), LTPR, LPPV	82,34	(ROURA et al., 2015)
Log-likelihood	79,99	(SOUPLET et al., 2008)
Trimmed Likelihood	79,09	(GARCÍA-LORENZO et al., 2011)

6.2 FUNÇÃO OBJETIVO PARA SEGMENTAÇÃO DE ESCLEROSE MÚLTIPLA NA BASE DE DADOS MICCAI2016

A configuração experimental inclui o uso do conjunto de dados MICCAI2008 e definição do alvo de aprendizagem correspondente na Seção 6.2.1.1, além da descrição do modo de avaliação e da infraestrutura computacional do MICCAI2016 na Seção 6.2.1.2. A apresentação dos resultados contém uma comparação do desempenho de segmentação entre diferentes algoritmos no MICCAI2016, na Seção 6.1.2.4.

6.2.1 Configuração Experimental

6.2.1.1 Conjunto de Dados e Alvo de Aprendizagem

A configuração experimental é semelhante à apresentada na Seção 6.1.1, usando as mesmas etapas de pre e posprocessamento, métodos classificação e treinamento. Mas, os experimentos desta seção empregam o conjunto de dados da competição de segmentação MICCAI2016. Assim, a função objetivo específica tem o 2D-score alterado para corresponder as métricas de segmentação dessa competição. Usamos uma combinação de três métricas: uma métrica por pixel *DSC* (ρ_1); e duas métricas de contagem por lesão, *LTPR* (ρ_2) e *LPPV* (ρ_3). Assim, para o MICCAI2016, a função objetivo específica emprega:

$$\text{2D-Score} = (\rho_1 + \rho_2 + \rho_3)/3. \tag{6.1}$$

Note que no MICCAI2016, a comparação de algoritmos ocorre através das métricas DSC, F1-score e ASD. Na Equação 6.1, enquanto o DSC aparece explicitamente, o F1-score é função de LTPR e LPPV sendo representado por estas. O ASD não foi otimizado nos experimentos aqui mencionados. Além disso, empregamos um 2D-score-3 com 3 fatias mediais: 85, 90 e 95, no plano axial do espaço MNI152.

6.2.1.2 Modo de Avaliação e Infraestrutura Computacional na Nuvem

A avaliação no MICCAI2016 foi realizada por meio de nossa participação nessa competição de segmentação de EM. Nosso método usando uma função objetivo específica, identificado como "**Equipe 9**", foi comparado com outros 12 algoritmos de segmentação de EM (Quadro 2). Cada participante gerou uma imagem binária do seu método de segmentação (um contêiner Docker) para rodar na infraestrutura computacional provida pela France Life Imaging platform. Assim, os competidores não tem acesso as imagens usadas para avaliação. Isto evita que algum competidor otimize o seu método de segmentação para cada paciente.

Quadro 2 – Algoritmos de Segmentação de EM Avaliados no MICCAI2016

Equipe	Referência	Algoritmo de Segmentação	Platforma
1	(BEAUMONT; COMMOWICK; BARILLOT, 2016)	*Graph cut*	CPU
2	(KARPATE; COMMOWICK; BARILLOT, 2015)	Detecção de anormalidades multi-modal	CPU
3	(FORBES et al., 2010)	*Markov Random Field* com dados poderados	CPU
4	(KHADEMI; VENETSANOPOULOS; MOODY, 2014)	Modelo baseado em borda	CPU
5	(MAHBOD; WANG; SMEDBY, 2016)	MLP com características espaciais	CPU
6	(MCKINLEY et al., 2016)	Ensemble de três 2D *Fully* CNN	GPU
7	(MUSCHELLI et al., 2016)	*Random Forest*	CPU
8	(CABEZAS et al., 2014)	Detecção de *outliers*	CPU
9	(SANTOS et al., 2016a)	MLP-PSO com função objetivo específica	CPU
10	(TOMAS-FERNANDEZ; WARFIELD, 2015)	Distribuição espacial e populacional	CPU
11	(URIEN et al., 2017)	*Max-tree* com contexto espacial	CPU
12	(VALVERDE et al., 2017)	Duas CNNs em Cascata	GPU
13	(VERA-OLMOS; MELERO; MALPICA, 2016)	*Random Forest* e *Markov Random Field*	CPU

6.2.2 Resultados

6.2.2.1 Comparação de Algoritmos de Segmentação no MICCAI2016

Os métodos de segmentação foram avaliados e agrupados conforme as métricas Dice e F1-score em: algoritmos não tão bons (vermelho), algoritmos de melhor desempenho

(verde) e especialista humanos (azul), vide Figura 15. O principal resultado é que todos os algoritmos avaliados no MICCAI2016 apresentaram segmentação inferior aos especialistas humanos. Nosso método (*Team 9*), apresentou desempenho de segmentação inferior a várias outras abordagens com Dice=0,34 e F1-score=0,168, mesmo usando uma função objetivo específica para a competição. Vale destacar ainda que nosso método usando um simples perceptron multicamada resultou no segundo menor tempo de segmentação por paciente de 6,42 minutos e com a menor quantidade de memória alocada 641,79 MB [1].

Observando os melhores resultados, o *Team 12* obteve o melhor F1-score=0,49 (com Dice=0,541) empregando duas redes convolucionais em cascata (VALVERDE et al., 2017). Além disso, o *Team 6* obteve o melhor Dice=0,591 (com F1-score=0,386) usando um comitê com três redes completamente convolucionais (MCKINLEY et al., 2016). Esses resultados nos motivaram a desenvolver métodos de segmentação de EM baseados em redes neurais convolucionais.

Além disso, um trabalho recente por Galassi et al. (2019) reportou Dice=0,624 e F1-score=0,604 avançando ligeiramente o estado-da-arte no conjunto de dados MICCAI2016, destacando que este resultado foi obtido após a respectiva competição. Tal abordagem foi baseada no modelo de rede neural de Valverde et al. (2017) treinado com o índice de Dice generalizado como função objetivo específica para segmentação.

[1] Dados suplementares da competição de segmentação de EM do MICCAI2016 disponíveis em <https://doi.org/10.5281/zenodo.1307653> último acesso em 13/12/2019.

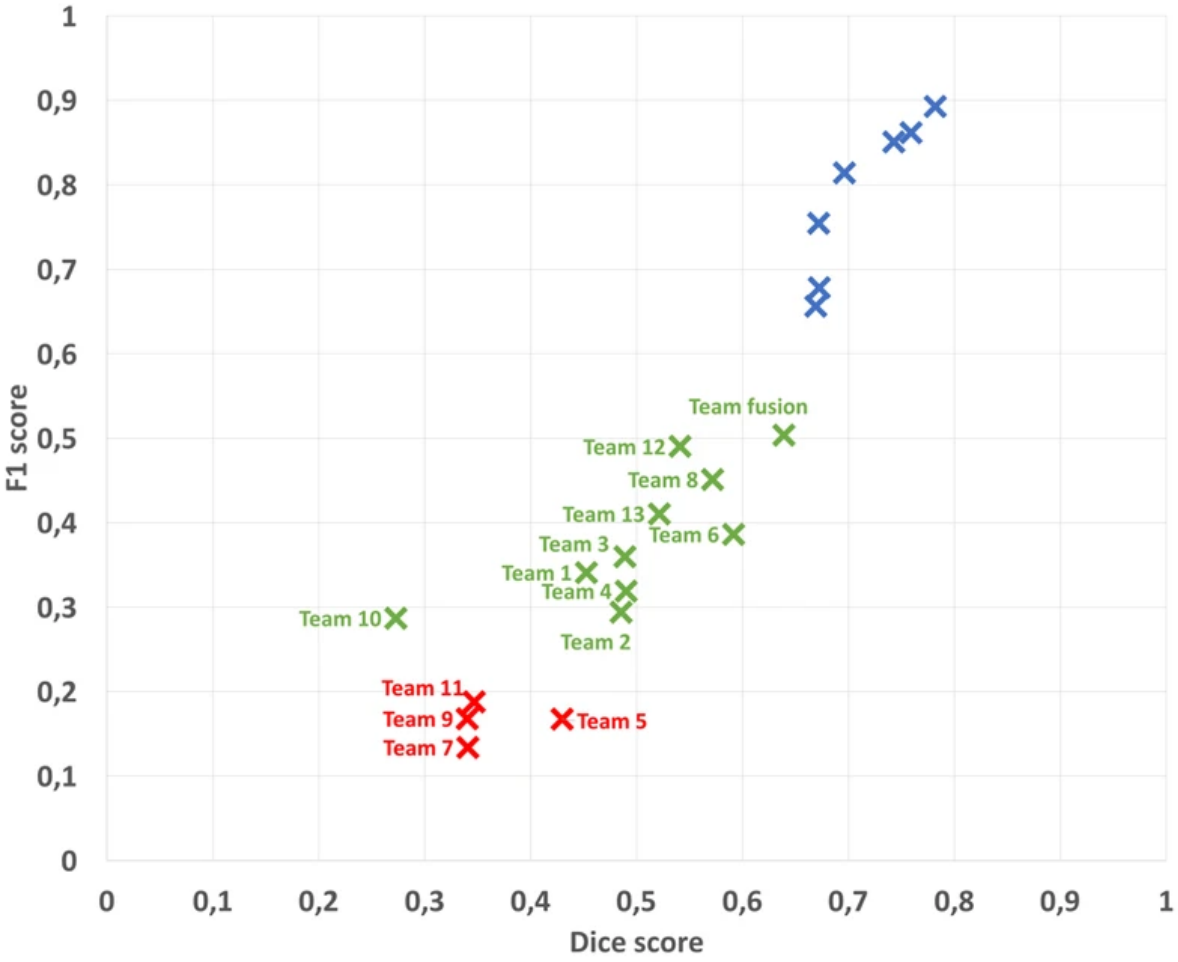

Figura 15 – Desempenho de segmentação com MLP-PSO usando função objetivo específica no MICCAI2016. Os métodos de segmentação foram avaliados com as métricas Dice (*score*) e F1-score. As cruzes no gráfico representam: algoritmos de pior segmentação (vermelho); algoritmos com melhor segmentação (verde); especialistas humanos (azul). Nosso método corresponde ao "*Team 9*". Fonte: (COMMOWICK et al., 2018).

6.3 MÁQUINA DE APRENDIZADO EXTREMO CONVOLUCIONAL PARA RECONHECIMENTO DE DÍGITOS NA BASE DE DADOS EMNIST

Nesta Seção descrevemos experimentos resultados relacionados ao desenvolvimento e ajuste da rede CELM com dois estágios de extração de características. Embora a aplicação primária seja a segmentação de lesões de EM, nós desenvolvemos nossos estudos aplicando as redes CELMs ao problema de reconhecimento de dígitos que é computacionalmente mais simples.

Nós iniciamos com a descrição da configuração experimental, na Seção 6.3.1. Em seguida, tratamos da seleção de filtros convolucionais para o primeiro (Seção 6.3.2) e o segundo estágio (Seção 6.3.3) de extração de características. Depois, ajustamos e validamos um modelo do erro de generalização contra o erro empírico obtido ao aplicar a

CELM com dois estágios, na Seção 6.3.4. Por fim, comparamos a CELM proposta com o Estado-da-Arte em reconhecimento de dígitos, na Seção 6.3.5.

6.3.1 Configuração Experimental

6.3.1.1 Especificação da Arquitetura de Rede

Os parâmetros usados nos dois estágios de extração de características estão na Tabela 8. No primeiro estágio, os parâmetros tamanho de filtro (W), tamanho de pooling (Q) e fator de subamostragem (D) foram definidos conforme o encontrado em (MCDONNELL; VLADUSICH, 2015). No segundo estágio, os valores desses parâmetros foram adaptados para as imagens de entrada com tamanho reduzido na entrada deste estágio, devido a subamostragem no primeiro estágio. Nas camadas de pooling, adotamos parâmetro $P = 2$ e kernel Gaussiano com desvio-padrão $\sigma = \sqrt{Q/2}$, similar a (SERMANET; CHINTALA; LECUN, 2012). O número de filtros de cada tipo é indicado antes do nome do filtro e foi definido por meio de validação cruzada com o método *repeated holdout*. Os filtros Gabor foram gerados variando a orientação em passos iguais, com o comprimento de onda definido como 6 para o primeiro estágio de extração de características e como 3 para o segundo estágio.

Tabela 8 – Parâmetros Usados na Extração de Características

Parâmetros	1º Estágio	2º Estágio
Tamanho de Filtro (W)	7	5
Tamanho de *Pooling* (Q)	7	4
Fator de Subamostragem (D)	3	2

No classificador, o parâmetro C da regressão rígida segue a heurística em (MCDONNELL; VLADUSICH, 2015):

$$C = \left(\frac{N}{M}\right)^2 \min(\text{diag}(\mathbf{A}\mathbf{A}^T)). \tag{6.2}$$

Além disso, a função de ativação é definida por $\phi(\mathbf{W}, \mathbf{F}) = (\mathbf{W}\mathbf{F})^2$, com o quadrado sendo computado por elemento. Para o cálculo da matriz de pesos de saída, apenas a Equação (5.7) foi empregada, pois é adequada para grandes conjuntos de treinamento.

Para a comparação de filtros, empregamos o método *repeated holdout* com 10 repetições e o conjunto de treinamento EMNIST-Digits dividido em duas partições: 90% para treinamento e 10% para validação. Neste caso, o tamanho máximo do conjunto de treinamento foi de 216.000 exemplos e o número de neurônios ocultos foi definido para 24.000.

6.3.2 Filtros no Primeiro Estágio de Extração de Características

Com objetivo de determinar uma composição para o banco de filtros no primeiro estágio convolutivo, avaliamos combinações de filtros, bem como filtros puros. A Figura

16 apresenta um gráfico de caixas com o erro de validação para filtros puros e combinados. O que pode ser visto claramente nessa figura é o baixo erro dos filtros combinados em relação aos filtros puros. Também nessa figura, os filtros aleatórios apresentaram o maior erro médio (círculos azuis).

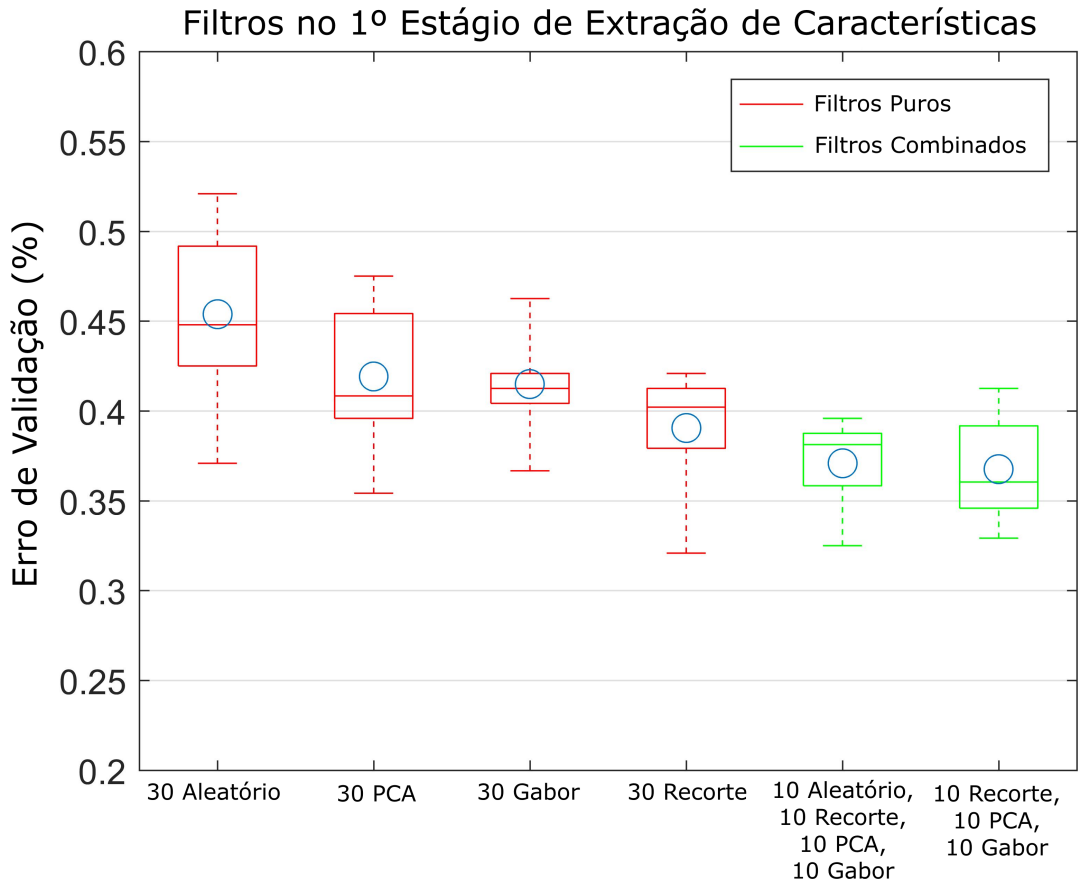

Figura 16 – Erro de generalização com filtros combinados versus puros em um único estágio de extração da máquina de aprendizado extremo convolucional. O teste de Friedman indicou a presença de diferença significativa nos erros entre bancos de filtros.

Em seguida, nós testamos a hipótese H_0 de que os diferentes bancos de filtros tem a mesma distribuição do erro de validação. O teste de Friedman levou a rejeição da hipótese de igualdade das distribuições do erro, resultando em p-valor igual a 0,0003, isto é, abaixo do nível de significância convencional de 5%. Em seguida, aplicamos o teste *post-hoc* de Conover com procedimento de correção de Holm. Os p-valores resultantes do teste *post-hoc* estão na Tabela 9, em negrito quando significativos. Observamos que as combinações de filtros foram significativamente melhores em relação aos filtro puros. Embora a diferença não tenha sido significativa, o erro de validação com os 30 filtros recorte foi marginalmente maior em comparação com as combinações de filtros. Entre as duas combinação de filtros avaliadas, também não houve diferença significativa.

Tabela 9 – Teste Post-hoc entre Pares de Bancos de Filtros Puros versus Bancos com Filtros Combinados.

	Puro				*Combinado*
	30 Aleatório	30 PCA	30 Gabor	30 Recorte	10 Aleatório, 10 Recorte, 10 PCA, 10 Gabor
Puro					
30 PCA	0,518				
30 Gabor	0,164	0,518			
30 Recorte	**0,005**	0,222	0,518		
Combinado					
10 Aleatório, 10 Recorte, 10 PCA, 10 Gabor	< **0,001**	< **0,001**	< **0,001**	0,055	
10 Recorte, 10 PCA, 10 Gabor	< **0,001**	**0,005**	**0,028**	0,518	0,518

A combinação {10 Patch, 10 PCA, 10 Gabor} resultou em um erro médio de validação de 0,37%, o menor entre os bancos de filtros testados usando um único estágio convolutivo. Por essa razão, nós mantemos essa combinação de filtros no primeiro estágio convolutivo, enquanto nós avaliamos filtros para o segundo estágio de extração, na próxima seção.

6.3.3 Filtros no Segundo Estágio de Extração de Características

Comparamos bancos de filtros para o segundo estágio convolutivo da CELM, a fim de determinar uma boa composição para tal banco de filtros. O primeiro estágio permaneceu fixado conforme a seção 6.3.2.

Filtros puros e combinados são comparados através do erro de validação, na Figura 17. Em geral, as médias dos erros (círculos azuis) foram próximas, ao redor de 0,3%, com o filtro aleatório apresentando o maior erro médio 0,33%. O teste de Friedman levou a não rejeição da hipótese de igualdade das distribuições (p-valor = 0,24). Como não obtivemos diferença estatisticamente significativa, adotamos o menor erro médio dentre os bancos com menor número de filtros como critério de seleção. Assim, selecionamos para o segundo estágio convolutivo da CELM a opção {6 Gabor} que resultou em erro médio de 0,30%. Além disso, o erro com 2 estágios convolutivos foi significativamente menor que com apenas 1 estágio, através do teste de Wilcoxon (p-valor = 0,003). A inclusão de um terceiro estágio de extração de características não reduziu ainda mais o erro.

Agregando os resultados desta Seção com os resultados da Seção 6.3.2, obtivemos a CELM ajustada com 2 estágios convolutivos e filtros selecionados (Figura 18).

6.3.4 Generalização Empírica versus Teórica

Nós estudamos o erro médio de validação em função do número de neurônios e de exemplos *conjuntamente*, com objetivo de descrever como esses dois fatores influenciam o erro médio no problema prático em questão. O perfil de erro para a rede CELM ajustada com 2 estágios convolutivos é apresentado na Figura 19. Em geral, podemos observar que

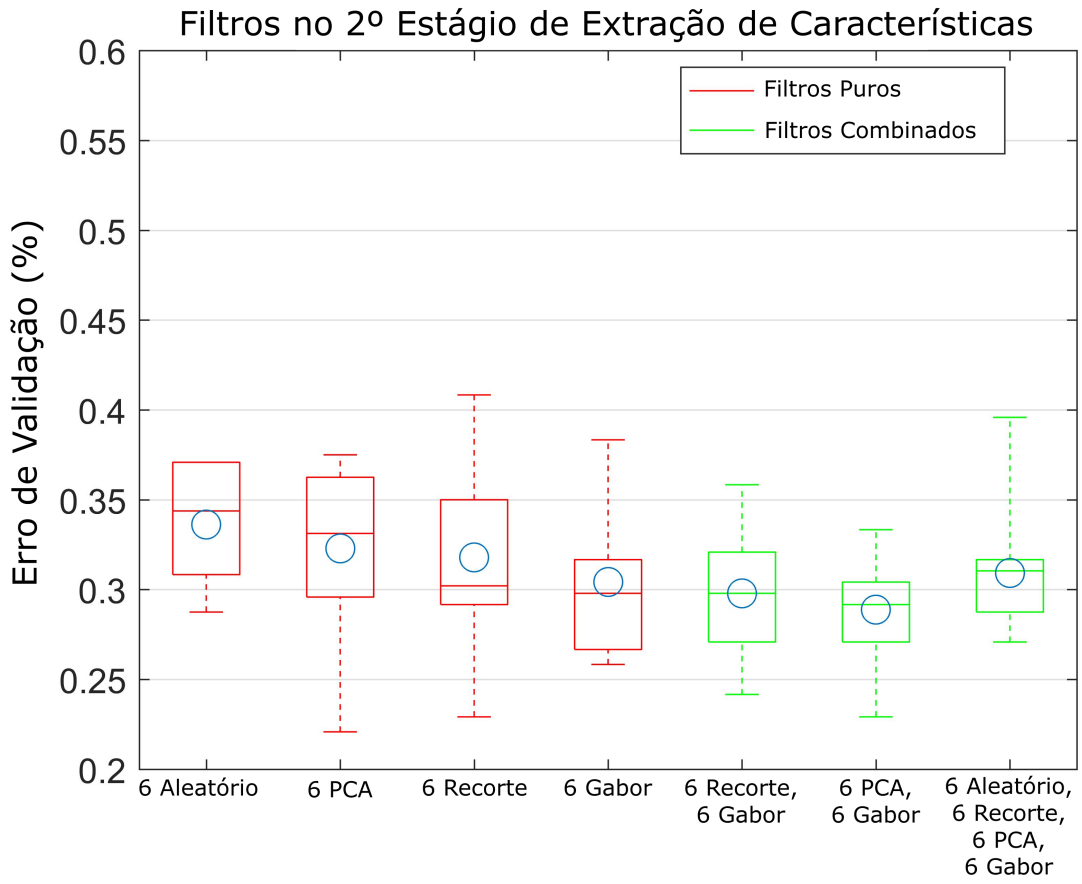

Figura 17 – Erro de generalização com filtros combinados versus puros no segundo estágio de extração da máquina de aprendizado extremo convolucional. O teste de Friedman não apontou diferenças significativas nos erros entre bancos de filtros.

há uma tendencia de decréscimo no erro médio com aumento do número de neurônios e, principalmente, de exemplos.

O erro de generalização médio empírico apresenta um padrão semelhante ao modelo de erro baseado no Teorema de Rahimi-Recht. Realizamos um ajuste não linear da superfície definida pelo erro teórico (Equação 5.14) contra o erro empírico. Então, os seguintes valores de coeficientes foram obtidos $C_K = 13,26$ e $C_M = 62,17$. O modelo ajustado (Figura 20) explica 76% da variação observada, de acordo com a estatística R-square ajustada, resultando em boa concordância. Assim, o modelo ajustado pode ser empregado para predição do erro de generalização.

Para avaliar a capacidade preditiva do modelo de erro ajustado, a aquisição de mais exemplos foi simulada e comparada com o erro predito. Dessa forma, todos os 240.000 exemplos disponíveis são usados para treinamento, em vez dos 216.000 exemplos usados para ajuste do modelo. Na Tabela 10, o erro médio observado no conjunto de testes foi de 0,225% com desvio padrão de 0,0095%, sendo próximo do erro previsto de 0,239%.

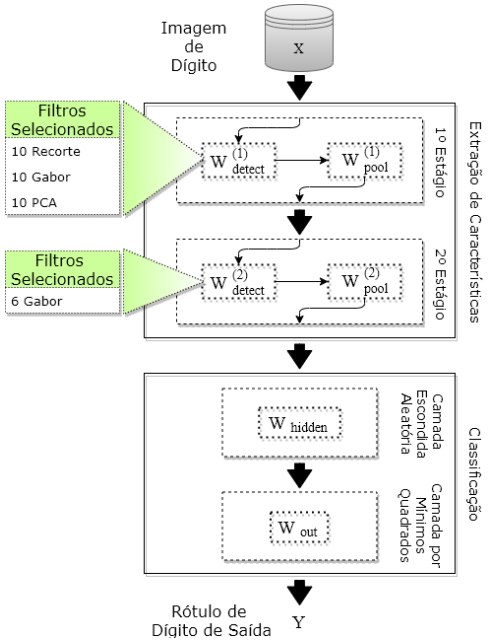

Figura 18 – Arquitetura da máquina de aprendizado extremo convolucional proposta com dois estágios convolucionais após a seleção de bancos de filtros.

Tabela 10 – Erro de Generalização Predito versus Observado

Erro Predito	Erro Observado
0,239%	$0,225\% \pm 0,0095\%$

Em um experimento adicional, substituímos o filtro gaussiano por um filtro de média na camada de pooling de ambos os estágios de extração. Com o filtro de média nas camadas de pooling da CELM de dois estágios, a taxa média de erro teve um leve aumento para 0,25%.

6.3.5 Comparação com o Estado-da-Arte no Reconhecimento de Dígitos

Comparamos a CELM proposta ao estado-arte em reconhecimento de dígitos. Primeiramente, consideramos para comparação apenas métodos baseados em redes neurais convolucionais de treinamento rápido, que são competitivas mas inferiores às abordagens baseadas em treinamento por retropropagação. Posteriormente, consideramos métodos de reconhecimento de dígitos em geral para comparação com nossa CELM. As abordagens foram separadas em três categorias: Não-CELM, Outras CELMs, Nossa CELM. Para métodos Não-CELM, coletamos resultados que foram relatados na literatura. Para as outras duas categorias, obtivemos resultados de experimentos próprios. Além disso, todos os 240.000 exemplos disponíveis no conjunto de treino são usados para treinamento e, portanto, não houve conjunto de validação.

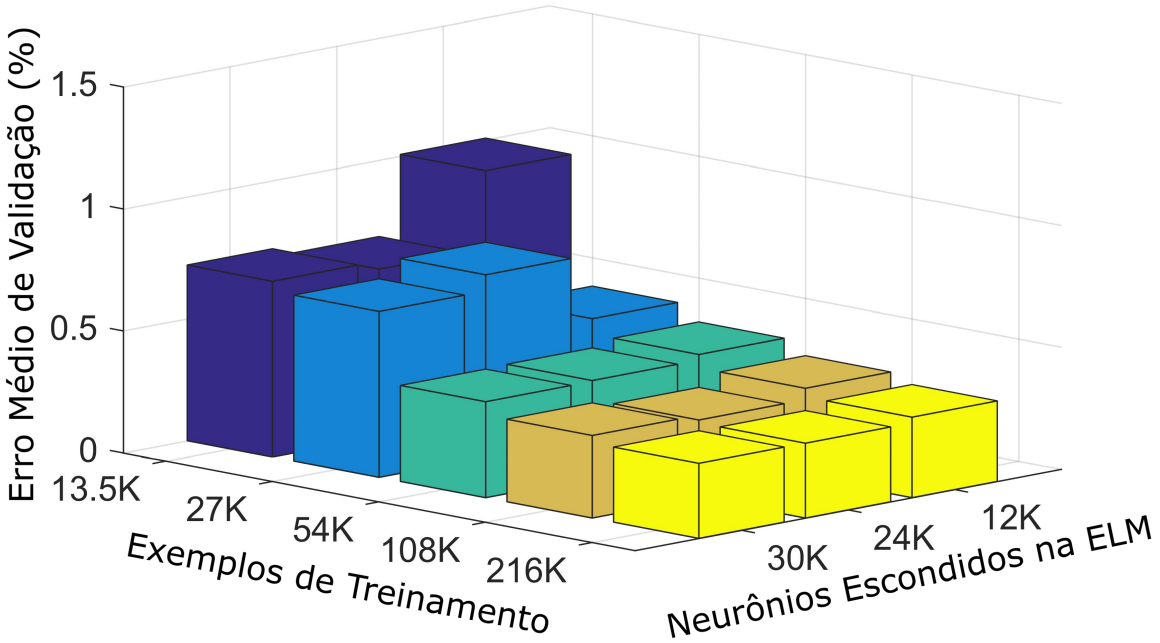

Figura 19 – Erro empírico para máquina de aprendizado extremo convolucional com dois estágios de extração de características e filtros selecionados. Confronte com a Figura 20.

Optamos por comparar os algoritmos principalmente sobre a base EMNIST, considerada mais desafiadora que o tradicional MNIST (Seção 3.2.3.1). A desvantagem desta escolha é que, como os resultados não são diretamente comparáveis entre diferentes conjuntos de dados, resultados de outras abordagens sobre o MNIST devem ser interpretados com cautela ao serem comparados com os resultados obtidos com a nossa CELM sobre o EMNIST. Note que este é o caso também da comparação entre EMNIST-digits-all e os dígitos do NIST-SD-19 que diferem pelo preprocessamento aplicado ao primeiro (BALDOMINOS; SAEZ; ISASI, 2019).

Para os experimentos, definimos 30.000 neurônios ocultos no estágio de classificação das CELMs. A comparação é realizada em dois conjuntos de dados: EMNIST-digits-balanced e EMNIST-digits-all. As CELMs foram treinadas com todos os exemplos disponíveis em cada conjunto de treinamento. Nas CELMs com 1 estágio de extração, definimos o número de filtros para 50. A rede LRF-ELM, por construção, não permite definir o número de neurônios de saída. Apesar disso, observe que com 50 filtros de tamanho 4×4 cada, o número de neurônios de saída é $50(28 - 4 + 1)^2 = 31.250$ e, portanto, próximo

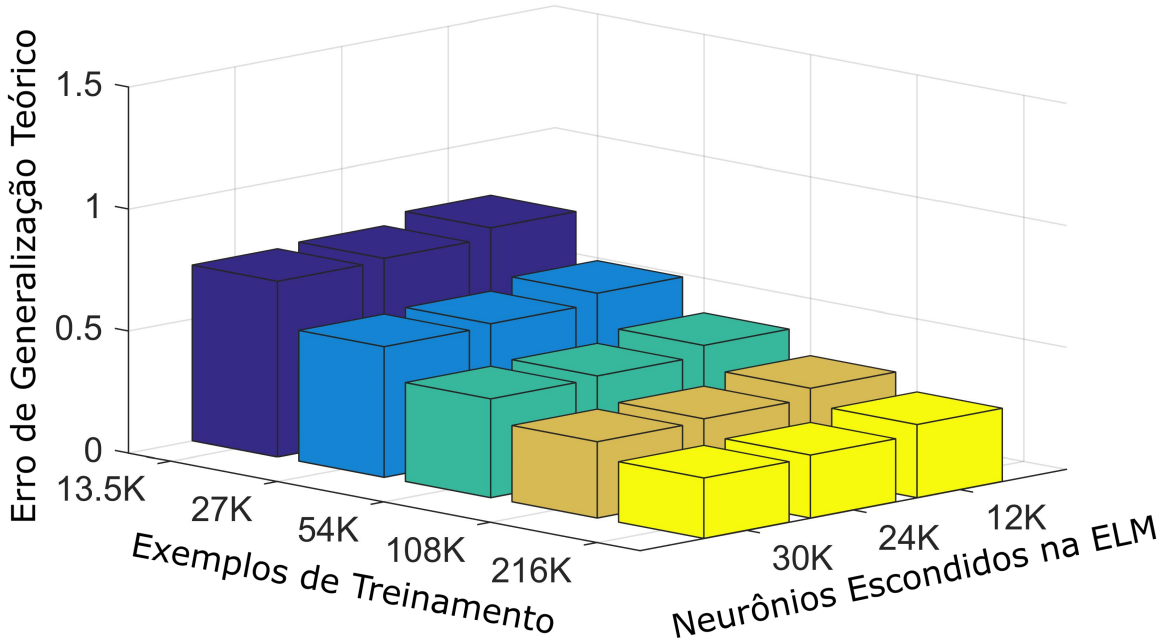

Figura 20 – Modelo de erro de generalização de Rahimi-Recht ajustado ao erro experimental que é mostrado na Figura 19.

ao número de neurônios ocultos definidos para outras CELMs avaliadas em nossos experimentos.

A Tabela 11 fornece taxas de erro de teste e tempos de treinamento médios, além dos tipos de computação para as três categorias de métodos de reconhecimento no conjunto de dados EMNIST-digits-balanced. A nossa abordagem com extração em 2 estágios obteve o menor erro 0,225% entre as CELMs. As abordagens baseadas em CNNs de (SHAWON et al., 2018) e (JAYASUNDARA et al., 2019) obtiveram erro ligeiramente menor 0,21%. Note que estes dois trabalhos foram publicados, respectivamente, em 03/12/2018 e 07/03/2019, portanto, após o nosso artigo (SANTOS; FILHO; SANTOS, 2019) publicado em 05/11/2018 que detinha o mais baixo erro conhecido no conjunto de dados EMNIST-digits-balanced. Nossas CELMs de um estágio com filtros 50-Patch ou 50-Gabor produziram erros de teste de 0,32 % e 0,33 %, respectivamente. Como esperado, essas taxas de erro foram muito semelhantes a outros CELMs de um estágio usando filtros 50-Patch (MCDONNELL; VLADUSICH, 2015) e 50-Kmeans de (YOUSEFI-AZAR; MCDONNELL, 2017) que obtiveram, respectivamente, 0,31% e 0,32% no erro de teste. No entanto, nossas implementações da CELM com um estágio foram um pouco mais rápidas, provavelmente devido ao uso de

laços executados em paralelo sobre os filtros na extração de características. Além disso, as CELMs utilizando unidade central de processamento (CPU) obtiveram tempos de treinamento competitivos (cerca de minutos) que são comparáveis ao tempo de treinamento de abordagens Não-CELM usando aceleração de unidade de processamento gráfico (GPU) (por exemplo (SINGH; PAUL; ARUN, 2017; SHAWON et al., 2018; JAYASUNDARA et al., 2019)).

Tabela 11 – Comparação com o Estado-da-Arte no EMNIST-digits-balanced

	Abordagem	Filtros Convolutionais Para Detecção	Tempo de Treinamento (Processamento)	Erro de Teste (%)
Não-CELM	Cohen et al. (2017)	-	20 horas (CPU)	4,10 ± 0,40
	Dufourq e Bassett (2017)	Ajuste Fino	1 dia (GPU)	0,70 ± 0,10
	Singh, Paul e Arun (2017)	Ajuste Fino	2 horas (CPU) 8 min (GPU)	0,38
	Shawon et al. (2018)	Ajuste Fino	11 min (GPU) *	**0,21**
	Jayasundara et al. (2019)	Ajuste Fino	5,5 horas (GPU) **	**0,21** ± 0,11
	Ma et al. (2019)	Ajuste Fino & Algoritmo Genético	18 dias (GPU)***	0,25
Outras CELMs	Huang et al. (2015b)	50-Random	**5 min** (CPU)	1,24 ± 0,03
	Yousefi-Azar e McDonnell (2017)	50-Kmeans	9 min (CPU)	0,32 ± 0,01
	McDonnell e Vladusich (2015)	50-Patch	9 min (CPU)	0,31 ± 0,01
Nossas CELMs	Filtros Puros Extração de 1-Estágio	50-PCA	7 min (CPU)	0,37 ± 0,01
		50-Random	8 min (CPU)	0,36 ± 0,01
		50-Gabor	7 min (CPU)	0,33 ± 0,01
		50-Patch	7 min (CPU)	0,32 ± 0,01
	Filtros Combinados Extração de 2-Estágios	1º estágio {10-Gabor,10-PCA,10-Patch} 2º estágio {6-Gabor}	9 min (CPU)	0,225 ± 0,01

* Não mencionado em (SHAWON et al., 2018), estimamos o tempo de treinamento, tomando $27s$ por época × 26 épocas resultando em cerca de 11 min, a partir de dados publicados por um dos autores em <https://github.com/jamilruet13> (último acesso em 13/12/2019).
** Não mencionado em (JAYASUNDARA et al., 2019), tempo de treinamento estimado tomando 0,916ms por imagem × 240.000 imagens × 90 épocas resultando em cerca de 5,5 horas, a partir de dados reportados por um dos autores para o MNIST (55s por 60.000 imagens em 1 época) em <https://github.com/vinojjayasundara/CapsNet-Keras> (último acesso em 13/12/2019).
*** Tempo reportado para uma única geração do algoritmo genético.

Já para a base de dados não-balanceada EMNIST-digits-all, comparamos a CELM proposta com outros trabalhos na Tabela 12. A nossa CELM com extração em 2 estágios obteve o menor erro em relação aos outros trabalhos. Além disso, o tempo de treinamento da CELM em CPU foi muito mais rápido que os demais trabalhos que usaram aceleração em GPU. Contudo, esses resultados são limitados pelo pequeno número de trabalhos reportando o erro sobre o EMNIST-digits-all ou, similarmente, sobre todos os dígitos do NIST-SD-19. Além disso, os outros trabalhos comparados não são atuais, o que significa que tempos de treinamento mais rápidos poderiam ter sido obtidos usando uma GPU mais recente.

Por último, consideramos trabalhos que empregaram a base de dados MNIST. Uma comparação fundada em bases de dados diferentes é sempre limitada, mas serve como referência, especialmente em relação ao erro humano de cerca de 0,2% (LECUN et al., 1995). Para o MNIST, o menor erro conhecido é 0,21%, porém usando um comitê de 5 redes (WAN et al., 2013). Se considerarmos uma única rede, o erro vai para 0,28%. A CELM proposta alcançou o menor erro conhecido com uma só rede.

Tabela 12 – Comparação com o Estado-da-Arte sobre o EMNIST-digits-all

Referência	Tempo de Treinamento (Processamento)	Erro de Teste (%)
(FORSTER; LÜCKE, 2017)	não reportado (GPU)	4,55 ± 0,01
(CIREŞAN; MEIER; SCHMIDHUBER, 2012)	alguns dias (GPU)	0,77
(CIREŞAN et al., 2011)	1-8 dias (GPU)	0,81 ± 0,02
CELM Proposta	**21 min** (CPU)	**0,28** ± 0,01

7 CONCLUSÃO

A presente tese teve como objetivo geral investigar redes neurais artificiais que apresentassem, idealmente, correção equivalente ao nível humano e treinamento rápido, em duas tarefas de classificação de padrões de imagens: a segmentação de lesões de esclerose múltipla e o reconhecimento de dígitos.

Retomando o problema de pesquisa, as redes neurais MLP e CNN representam pontos de partida opostos para lidar com a classificação de padrões de imagens. A rede MLP pode ser treinada rapidamente com o algoritmo PSO (em minutos), mas resultando em acurácia insuficiente. Por outro lado, a CNN representa o estado-da-arte em diversas tarefas de classificação de padrões de imagens, mas o treinamento é lento (em horas-dias). Este estudo investigou, avaliou e propôs algoritmos visando resolver as limitações desses dois tipos de redes neurais.

A partir deste estudo podemos concluir que:

- métricas de segmentação na função objetivo específica para segmentação de EM podem resultar em melhor segmentação observada;
- máquinas de aprendizado extremo convolucionais podem atingir erro de teste do estado-da-arte de $0,225\%$ e próximo ao desempenho humano de $0,2\%$ (segundo (LECUN et al., 1995)) em reconhecimento de dígitos, enquanto mantêm o treinamento rápido.

No restante deste capítulo: sumarizamos e avaliamos os resultados obtidos, na Seção 7.1; discutimos limitações e implicações, na Seção 7.2; apontamos trabalhos futuros, na Seção 7.3; e destacamos as principais contribuições, na Seção 7.4. Por fim, na Seção 7.5, listamos os trabalhos publicados no decorrer do doutorado que serviram de base para a presente tese.

7.1 SUMÁRIO DOS RESULTADOS

Na abordagem com rede neural MLP, este estudo propôs funções objetivo específicas para segmentação de lesões de EM. As funções objetivo específicas foram propostas para tratar o problema da divergência entre função objetivo e métricas de segmentação, que ocorre particularmente em abordagens de segmentação de EM.

Apontamos a inviabilidade de usar métricas de segmentação tridimensionais custosas diretamente como função objetivo em treinamento iterativo. Então, propusemos a substituição de métricas tridimensionais por métricas mais rápidas, computadas por fatia, na função objetivo específica.

Este estudo indicou que o desempenho de segmentação da função objetivo específica foi superior (3D-Score=83,26%) ao da função objetivo com MSE (3D-Score=56,30%), no conjunto de dados MICCAI2008. Além disso, apresentamos evidências de que o desempenho de segmentação com a função objetivo específica é comparável ou intermediário em relação ao estado-da-arte, respectivamente, nos conjunto de dados MICCAI2008 e MICCAI2016. Em geral, esses resultados foram considerado pouco satisfatórios o que nos levou a investigar outro modelo de classificador neural.

Já em uma segunda abordagem, investigamos CELMs, uma alternativa para desenvolver uma CNN de treinamento rápido e com elevada acurácia, abordando a tarefa de reconhecimento de dígitos. Propomos uma CELM profunda com combinação de filtros, dois estágios de extração de características e duas camadas no estágio de classificação. Comparamos bancos de filtros puros e combinados, em dois estágios de extração de características da CELM.

Este estudo mostrou que nem sempre o erro de validação foi menor com os filtros combinados do que com os filtros puros. Em nossa análise, selecionamos um banco de filtros combinados, no primeiro estágio de extração, e um banco de filtros puros, no segundo estágio de extração. Empregamos a CELM com filtros selecionados para investigar o erro de generalização.

Este estudo também mostrou que o modelo de erro baseado no teorema de Rahimi-Retch é capaz de ajustar e prever o erro empírico, com diferença de 0,014% entre erro de teste observado e predito. Deste modo, contribuímos para um melhor entendimento do erro de generalização em função do tamanho conjunto de treinamento e do número de parâmetros, dois fatores que eram apenas conjecturados como razões do sucesso do aprendizado profundo.

Comparamos a CELM proposta com algoritmos do estado-da-arte em reconhecimento de dígitos. A CELM proposta resultou em acurácia superior (99,775%), bem como tempo de treinamento competitivo (9 min, CPU), mesmo em relação a abordagens que empregam processamento em GPUs, no conjunto de dados de reconhecimento de dígitos EMNIST .

7.2 DISCUSSÃO

Podemos apontar algumas limitações em nosso estudo. Na abordagem com MLP para segmentação de lesões de EM, a simplicidade da rede empregada, com uma camada oculta e baixo número de pesos, pode ter afetado negativamente o desempenho de segmentação. Outra limitação, na função objetivo específica, usamos métricas bidimensionais com poucas fatias e, portanto, com baixa fidelidade (correlação) em relação as métricas 3D. A baixa fidelidade das métricas bidimensionais na função objetivo pode levar a maior variabilidade no desempenho de segmentação observado, comprometendo a robustez do método de segmentação. Já na quanto a CELM proposta, a principal limitação está na grande quantidade de memória requerida que tende a crescer com o número de neurônios

escondidos, tamanho do conjunto de treino e o tamanho das imagens de entrada. Outra limitação, no ajuste do modelo de erro, não avaliamos intervalos de confiança na predição do erro.

Algumas implicações práticas deste estudo vêm da possibilidade de predizer o erro empírico em redes CELM, a partir do ajuste do modelo de erro baseado no Teorema de Rahimi-Retch. A predição do erro de generalização tem aplicação, por exemplo, no projeto e gerenciamento de sistemas de reconhecimento para alocação de recursos computacionais ou para guiar a tomada de decisão sobre o tamanho do conjunto de dados (HESTNESS et al., 2017). O melhor desempenho de segmentação da função objetivo específica sobre a inespecífica indica que funções objetivo específicas podem ser usadas para aprimorar outros métodos de segmentação de lesões de EM. A aplicação de função objetivo específica em métodos de segmentação do estado-da-arte pode levar a um aprimoramento adicional do desempenho de segmentação de lesões de EM.

7.3 TRABALHOS FUTUROS

A seguir, apontamos possíveis tópicos para investigação adicional. Em nosso estudo da função objetivo para segmentação de EM, apenas métricas de segmentação de baixa fidelidade foram empregadas na função objetivo específica. Um progresso natural seria avaliar o desempenho de segmentação em função da fidelidade das métricas usadas na função objetivo específica. Em outra frente de pesquisa, trabalho adicional é necessário para avaliar a CELM na segmentação de lesões de EM.

7.4 CONTRIBUIÇÕES

Nesta seção declaramos as principais contribuições realizadas, a seguir

1. Proposta e avaliação das funções objetivo específicas para segmentação de lesões de EM com MLP, nas bases de dados MICCAI2008 e MICCAI20016.
2. Rede CELM profunda com combinação de filtros convolucionais.
3. Validação experimental do modelo de erro baseado no teorema de Rahimi-Retch.
4. Avaliação da CELM proposta quanto ao erro de teste no EMNIST.

7.5 PUBLICAÇÕES

Durante o doutorado tivemos trabalhos aceitos em conferências e periódicos, listados e contextualizados a seguir em ordem cronológica de publicação.

Trabalhos publicados em eventos:

- SANTOS, M. M.; DINIZ, P. R. B.; SILVA-FILHO, A. G.; SANTOS, W. P. Evaluation-oriented training via surrogate metrics for multiple sclerosis segmentation. In: OURSELIN, S.; JOSKOWICZ, L.; SABUNCU, M. R.; UNAL, G.; WELLS, W. (Ed.). *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*. Cham: Springer International Publishing, 2016. p. 398–405. ISBN 978-3-319-46723-8. (Apêndice A).

  Publicado na trilha principal do MICCAI2016, o primeiro artigo propõe uma função objetivo específica para segmentação de lesões de EM com uma simples rede perceptron multicamada sobre o conjunto de dados da competição de segmentação de EM do MICCAI2008. Em comparação com duas outras abordagens que empregaram redes convolucionais profundas, o treinamento com função objetivo específica do perceptron multicamada levou a um resultado de segmentação próximo ao de Brosch et al. (2016), mas que tornou-se posteriormente inferior ao de Valverde et al. (2017) (conforme Tabela 7). Além disso, comparando funções objetivo quanto ao desempenho de segmentação de EM, o treinamento com função objetivo específica superou o treinamento com a função inespecífica tendo o erro quadrático como alvo de aprendizagem (vide Tabela 6).

- SANTOS, M. M.; DINIZ, P.; SILVA-FILHO, A. G.; SANTOS, W. P. Evaluation-oriented training strategy on ms segmentation challenge 2016. In: COMMOWICK, O.; CERVENANSKY, F.; AMELI, R. (Ed.). *Proceedings of the 1st MICCAI Challenge on Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure–MICCAI-MSSEG*. [S.l.: s.n.], 2016. p. 57–62. (Apêndice B).

  Publicado no *workshop* da competição de segmentação de EM do MICCAI2016, este segundo artigo teve algumas diferenças em relação ao primeiro, mencionadas a seguir. Uma nova função objetivo com alvo de aprendizagem para métricas de segmentação de EM definidas nessa nova competição. Um novo conjunto de dados foi também abordado. Além disso, a participação nessa competição envolveu, além do segundo artigo mencionado, a elaboração de um *pipeline* de segmentação de EM para um ambiente de computação em nuvem e uma apresentação descrevendo o método de segmentação. Posteriormente, em coautoria com os demais participantes dessa competição, um artigo foi publicado em periódico, contendo os resultados da comparação das diversas abordagens de segmentação de EM concorrentes, conforme descrito a seguir.

Trabalhos publicados em periódicos:

- COMMOWICK, O.; ISTACE, A.; KAIN, M.; LAURENT, B.; LERAY, F.; SIMON, M.; POP, S. C.; GIRARD, P.; AMÉLI, R.; FERRÉ, J.-C.; KERBRAT,

A.; TOURDIAS, T.; CERVENANSKY, F.; GLATARD, T.; BEAUMONT, J.; DOYLE, S.; FORBES, F.; KNIGHT, J.; KHADEMI, A.; MAHBOD, A.; WANG, C.; MCKINLEY, R.; WAGNER, F.; MUSCHELLI, J.; SWEENEY, E.; ROURA, E.; LLADÓ, X.; SANTOS, M. M.; SANTOS, W. P.; SILVA-FILHO, A. G.; TOMAS-FERNANDEZ, X.; URIEN, H.; BLOCH, I.; VALVERDE, S.; CABEZAS, M.; VERA-OLMOS, F. J.; MALPICA, N.; GUTTMANN, C.; VUKUSIC, S.; EDAN, G.; DOJAT, M.; STYNER, M.; WARFIELD, S. K.; COTTON, F.; BARILLOT, C. Objective evaluation of multiple sclerosis lesion segmentation using a data management and processing infrastructure. *Scientific Reports*, v. 8, n. 1, p. 13650, 2018. ISSN 2045-2322. (Apêndice C).

O terceiro artigo, fruto de uma colaboração internacional que teve origem na competição de segmentação de EM do MICCAI2016, apresentou um estudo comparando diversos algoritmos de segmentação de lesões de EM. O principal resultado do trabalho foi que todos os algoritmos avaliados apresentaram segmentação inferior ao consenso entre especialistas humanos (vide Figura 15). Nosso método de segmentação com um perceptron multicamada simples, identificado por "Team 9", apresentou desempenho inferior a outras abordagens, mesmo com função objetivo específica para a competição. O melhor algoritmo, representado por "Team 12", empregou uma rede neural convolucional profunda. Esses resultados motivaram nossos estudos em redes neurais convolucionais, mas que pudessem ser treinadas rapidamente.

- SANTOS, M. M. dos; FILHO, A. G. da S.; SANTOS, W. P. dos. Deep convolutional extreme learning machines: Filters combination and error model validation. *Neurocomputing*, Elsevier, v. 329, p. 359–369, 2019. (Apêndice D).

  No quarto artigo estudamos a rede CELM, um tipo de rede neural convolucional de treinamento rápido, como alternativa às CNNs tradicionais que apresentam longo tempo de treinamento. Ao invés da tarefa de segmentação de lesões de EM, nós estudamos CELMs aplicadas ao reconhecimento de dígitos, por ser uma tarefa de classificação de padrões de imagens mais simples e tradicionalmente usada para avaliar redes convolucionais. Nós propomos uma CELM que emprega novas combinações de diversos tipos de filtros convolucionais, mais profunda e com menor erro que outras CELMs existentes. Além disso, mostramos que o erro de gerenalização para a CELM proposta pode ser predito pelo modelo de Rahimi-Retch, a partir do tamanho do conjunto de treinamento e do número de neurônios escondidos da ELM. A CELM proposta obteve o menor erro de classificação conhecido, na época da publicação, para o conjunto de dados de reconhecimento de dígitos EMNIST (conforme Tabela 11).

REFERÊNCIAS

AGRAWAL, P.; GIRSHICK, R.; MALIK, J. Analyzing the performance of multilayer neural networks for object recognition. In: SPRINGER. *European conference on computer vision*. [S.l.], 2014. p. 329–344.

AKHONDI-ASL, A.; HOYTE, L.; LOCKHART, M. E.; WARFIELD, S. K. A logarithmic opinion pool based staple algorithm for the fusion of segmentations with associated reliability weights. *IEEE transactions on medical imaging*, IEEE, v. 33, n. 10, p. 1997–2009, 2014.

ALSHAYEJI, M. H.; AL-ROUSAN, M. A.; ELLETHY, H.; ABED, S. An efficient multiple sclerosis segmentation and detection system using neural networks. *Computers & Electrical Engineering*, Elsevier, v. 71, p. 191–205, 2018.

BA, J.; CARUANA, R. Do deep nets really need to be deep? In: GHAHRAMANI, Z.; WELLING, M.; CORTES, C.; LAWRENCE, N. D.; WEINBERGER, K. Q. (Ed.). *Advances in Neural Information Processing Systems 27*. [S.l.]: Curran Associates, Inc., 2014. p. 2654–2662.

BA, J.; MNIH, V.; KAVUKCUOGLU, K. Multiple object recognition with visual attention. *arXiv preprint arXiv:1412.7755*, 2014.

BACH, F. On the equivalence between kernel quadrature rules and random feature expansions. *Journal of Machine Learning Research*, v. 18, n. 21, p. 1–38, 2017.

BALDOMINOS, A.; SAEZ, Y.; ISASI, P. A survey of handwritten character recognition with mnist and emnist. *Applied Sciences*, Multidisciplinary Digital Publishing Institute, v. 9, n. 15, p. 3169, 2019.

BARKHOF, F.; FILIPPI, M.; MILLER, D. H.; SCHELTENS, P.; CAMPI, A.; POLMAN, C. H.; COMI, G.; ADER, H. J.; LOSSEFF, N.; VALK, J. Comparison of mri criteria at first presentation to predict conversion to clinically definite multiple sclerosis. *Brain*, Oxford Univ Press, v. 120, n. 11, p. 2059–2069, 1997.

BEAUMONT, J.; COMMOWICK, O.; BARILLOT, C. Multiple sclerosis lesion segmentation using an automated multimodal graph cut. In: *Proceedings of the 1st MICCAI Challenge on Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure–MICCAI-MSSEG*. [S.l.: s.n.], 2016. p. 1–8.

BERNAL, J.; KUSHIBAR, K.; ASFAW, D. S.; VALVERDE, S.; OLIVER, A.; MARTí, R.; LLADó, X. Deep convolutional neural networks for brain image analysis on magnetic resonance imaging: a review. *Artificial Intelligence in Medicine*, v. 95, p. 64–81, abr. 2019. ISSN 0933-3657. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0933365716305206>.

BISHOP, C. M. *Neural Networks for Pattern Recognition*. 1. ed. Oxford University Press, USA, 1996. Paperback. ISBN 9780198538646. Disponível em: <http://www.worldcat.org/isbn/0198538642>.

BISHOP, C. M. *Pattern recognition and machine learning*. [S.l.]: Springer Science+ Business Media, 2006.

BRIDLE, J. S. Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In: SOULIé, F. F.; HéRAULT, J. (Ed.). *Neurocomputing - Algorithms, Architectures and Applications*. Springer Berlin Heidelberg, 1990, (NATO ASI Series, v. 68). p. 227–236. ISBN 978-3-642-76155-3. Disponível em: <http://dx.doi.org/10.1007/978-3-642-76153-9_28>.

BROSCH, T.; TANG, L. Y.; YOO, Y.; LI, D. K.; TRABOULSEE, A.; TAM, R. Deep 3d convolutional encoder networks with shortcuts for multiscale feature integration applied to multiple sclerosis lesion segmentation. *IEEE transactions on medical imaging*, IEEE, v. 35, n. 5, p. 1229–1239, 2016.

CABEZAS, M.; OLIVER, A.; ROURA, E.; FREIXENET, J.; VILANOVA, J. C.; RAMIÓ-TORRENTÀ, L.; ROVIRA, À.; LLADÓ, X. Automatic multiple sclerosis lesion detection in brain mri by flair thresholding. *Computer methods and programs in biomedicine*, Elsevier, v. 115, n. 3, p. 147–161, 2014.

CERASA, A.; BILOTTA, E.; AUGIMERI, A.; CHERUBINI, A.; PANTANO, P.; ZITO, G.; LANZA, P.; VALENTINO, P.; GIOIA, M. C.; QUATTRONE, A. A cellular neural network methodology for the automated segmentation of multiple sclerosis lesions. *Journal of neuroscience methods*, Elsevier, v. 203, n. 1, p. 193–199, 2012.

CHAN, T.-H.; JIA, K.; GAO, S.; LU, J.; ZENG, Z.; MA, Y. Pcanet: A simple deep learning baseline for image classification? *IEEE Transactions on Image Processing*, IEEE, v. 24, n. 12, p. 5017–5032, 2015.

CHAVHAN, G. B. *MRI made easy*. [S.l.]: JP Medical Ltd, 2013.

CHEN, S.; MONTGOMERY, J.; BOLUFÉ-RÖHLER, A. Measuring the curse of dimensionality and its effects on particle swarm optimization and differential evolution. *Applied Intelligence*, Springer, v. 42, n. 3, p. 514–526, 2015.

CIREŞAN, D.; MEIER, U.; SCHMIDHUBER, J. Multi-column deep neural networks for image classification. In: IEEE. *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. [S.l.], 2012. p. 3642–3649.

CIREŞAN, D. C.; MEIER, U.; GAMBARDELLA, L. M.; SCHMIDHUBER, J. Convolutional neural network committees for handwritten character classification. In: IEEE. *Document Analysis and Recognition (ICDAR), 2011 International Conference on*. [S.l.], 2011. p. 1135–1139.

COHEN, G.; AFSHAR, S.; TAPSON, J.; SCHAIK, A. van. Emnist: an extension of mnist to handwritten letters. *arXiv preprint arXiv:1702.05373*, 2017.

COLLINS, D. L.; NEELIN, P.; PETERS, T. M.; EVANS, A. C. Automatic 3d intersubject registration of mr volumetric data in standardized talairach space. *Journal of computer assisted tomography*, v. 18, n. 2, p. 192–205, 1994.

COLLINS, D. L.; WARD, G. *MINC Tools mni autoreg*. 2017. <https://en.wikibooks.org/wiki/MINC/Tools/mni_autoreg>, último acesso em 13/12/2019.

COMMOWICK, O.; ISTACE, A.; KAIN, M.; LAURENT, B.; LERAY, F.; SIMON, M.; POP, S. C.; GIRARD, P.; AMÉLI, R.; FERRÉ, J.-C.; KERBRAT, A.; TOURDIAS, T.; CERVENANSKY, F.; GLATARD, T.; BEAUMONT, J.; DOYLE, S.; FORBES, F.; KNIGHT, J.; KHADEMI, A.; MAHBOD, A.; WANG, C.; MCKINLEY, R.; WAGNER, F.; MUSCHELLI, J.; SWEENEY, E.; ROURA, E.; LLADÓ, X.; SANTOS, M. M.; SANTOS, W. P.; SILVA-FILHO, A. G.; TOMAS-FERNANDEZ, X.; URIEN, H.; BLOCH, I.; VALVERDE, S.; CABEZAS, M.; VERA-OLMOS, F. J.; MALPICA, N.; GUTTMANN, C.; VUKUSIC, S.; EDAN, G.; DOJAT, M.; STYNER, M.; WARFIELD, S. K.; COTTON, F.; BARILLOT, C. Objective evaluation of multiple sclerosis lesion segmentation using a data management and processing infrastructure. *Scientific Reports*, v. 8, n. 1, p. 13650, 2018. ISSN 2045-2322.

COMPSTON, A.; COLES, A. Multiple sclerosis. *The Lancet*, Elsevier, v. 372, n. 9648, p. 1502–1517, Oct 2008. ISSN 0140-6736. Disponível em: <https://doi.org/10.1016/S0140-6736(08)61620-7>.

CONSTANTINIDES, C. *Magnetic resonance imaging: the basics*. [S.l.]: CRC press, 2016.

DAUGMAN, J. G. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of the Optical Society of America, A, Optics, Image & Science*, Optical Society of America, 1985.

DICE, L. R. Measures of the amount of ecologic association between species. *Ecology*, Wiley Online Library, v. 26, n. 3, p. 297–302, 1945.

DING, S.; GUO, L.; HOU, Y. Extreme learning machine with kernel model based on deep learning. *Neural Computing and Applications*, Springer, v. 28, n. 8, p. 1975–1984, 2017.

DUFOURQ, E.; BASSETT, B. A. Eden: Evolutionary deep networks for efficient machine learning. *arXiv preprint arXiv:1709.09161*, 2017.

EBERHART, R. C.; SHI, Y. *Computational Intelligence - Concepts to Implementations*. [S.l.]: Elsevier, 2007. ISBN 978-1-55860-759-0.

EITEL, A.; SPRINGENBERG, J. T.; SPINELLO, L.; RIEDMILLER, M.; BURGARD, W. Multimodal deep learning for robust rgb-d object recognition. In: IEEE. *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. [S.l.], 2015. p. 681–687.

ENGELBRECHT, A. P. *Computational intelligence: an introduction*. [S.l.]: John Wiley & Sons, 2007.

FORBES, F.; DOYLE, S.; GARCIA-LORENZO, D.; BARILLOT, C.; DOJAT, M. A weighted multi-sequence markov model for brain lesion segmentation. In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. [S.l.: s.n.], 2010. p. 225–232.

FORSTER, D.; LÜCKE, J. Truncated variational em for semi-supervised neural simpletrons. In: IEEE. *2017 International Joint Conference on Neural Networks (IJCNN)*. [S.l.], 2017. p. 3769–3776.

FRITSCHER, K.; RAUDASCHL, P.; ZAFFINO, P.; SPADEA, M. F.; SHARP, G. C.; SCHUBERT, R. Deep neural networks for fast segmentation of 3d medical images. In: SPRINGER. *International Conference on Medical Image Computing and Computer-Assisted Intervention*. [S.l.], 2016. p. 158–165.

GALASSI, F.; TARRIDE, S.; VALLÉE, E.; COMMOWICK, O.; BARILLOT, C. Deep learning for multi-site ms lesions segmentation: two-step intensity standardization and generalized loss function. In: *ISBI 2019 - 16th IEEE International Symposium on Biomedical Imaging*. Venice, Italy: IEEE, 2019. p. 1. Disponível em: <https://hal.archives-ouvertes.fr/hal-02052250>.

GAO, S.; ZHOU, M.; WANG, Y.; CHENG, J.; YACHI, H.; WANG, J. Dendritic neuron model with effective learning algorithms for classification, approximation, and prediction. *IEEE transactions on neural networks and learning systems*, IEEE, v. 30, n. 2, p. 601–614, 2018.

GARCÍA-LORENZO, D.; FRANCIS, S.; NARAYANAN, S.; ARNOLD, D. L.; COLLINS, D. L. Review of automatic segmentation methods of multiple sclerosis white matter lesions on conventional magnetic resonance imaging. *Medical image analysis*, Elsevier, v. 17, n. 1, p. 1–18, 2013.

GARCÍA-LORENZO, D.; PRIMA, S.; ARNOLD, D. L.; COLLINS, D. L.; BARILLOT, C. Trimmed-likelihood estimation for focal lesions and tissue segmentation in multisequence mri for multiple sclerosis. *Medical Imaging, IEEE Transactions on*, IEEE, v. 30, n. 8, p. 1455–1467, 2011.

GIRSHICK, R.; DONAHUE, J.; DARRELL, T.; MALIK, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2014. p. 580–587.

GONZALEZ, R. C.; WOODS, R. E. *Digital Image Processing (4nd Edition)*. [S.l.]: Pearson, 2018.

GOODFELLOW, I.; COURVILLE, A.; BENGIO, Y. *Deep learning*. [S.l.]: The MIT Press, 2016.

GRAHL, S.; PONGRATZ, V.; SCHMIDT, P.; ENGL, C.; BUSSAS, M.; RADETZ, A.; GONZALEZ-ESCAMILLA, G.; GROPPA, S.; ZIPP, F.; LUKAS, C. et al. Evidence for a white matter lesion size threshold to support the diagnosis of relapsing remitting multiple sclerosis. *Multiple sclerosis and related disorders*, Elsevier, v. 29, p. 124–129, 2019.

GUDISE, V. G.; VENAYAGAMOORTHY, G. K. Comparison of particle swarm optimization and backpropagation as training algorithms for neural networks. In: IEEE. *Swarm Intelligence Symposium, 2003. SIS'03. Proceedings of the 2003 IEEE*. [S.l.], 2003. p. 110–117.

GUIZARD, N.; COUPé, P.; FONOV, V. S.; MANJóN, J. V.; ARNOLD, D. L.; COLLINS, D. L. Rotation-invariant multi-contrast non-local means for {MS} lesion segmentation. *NeuroImage: Clinical*, v. 8, n. 0, p. 376 – 389, 2015. ISSN 2213-1582.

HAN, F.; HUANG, D.-S. Improved extreme learning machine for function approximation by encoding a priori information. *Neurocomputing*, Elsevier, v. 69, n. 16-18, p. 2369–2373, 2006.

HAUSER, S. L.; GOODKIN, D. Multiple sclerosis and other demyelinating diseases. *Harrisons principles of internal medicine*, v. 2, p. 2452–2461, 2001.

HAYKIN, S. *Redes Neurais: Princípios e Prática*. [S.l.]: Bookman, 2001.

HAYKIN, S. *Neural Networks and Learning Machines*. 3. ed. [S.l.]: Prentice Hall, 2008.

HE, L.; CHAO, Y.; SUZUKI, K. Two efficient label-equivalence-based connected-component labeling algorithms for 3-d binary images. *Image Processing, IEEE Transactions on*, IEEE, v. 20, n. 8, p. 2122–2134, 2011.

HE, L.; CHAO, Y.; SUZUKI, K.; WU, K. Fast connected-component labeling. *Pattern recognition*, Elsevier, v. 42, n. 9, p. 1977–1987, 2009.

HESTNESS, J.; NARANG, S.; ARDALANI, N.; DIAMOS, G.; JUN, H.; KIANINEJAD, H.; PATWARY, M.; ALI, M.; YANG, Y.; ZHOU, Y. Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*, 2017.

HUANG, G.; HUANG, G.-B.; SONG, S.; YOU, K. Trends in extreme learning machines: A review. *Neural Networks*, Elsevier, v. 61, p. 32–48, 2015.

HUANG, G.-B.; BAI, Z.; KASUN, L. L. C.; VONG, C. M. Local receptive fields based extreme learning machine. *IEEE Computational Intelligence Magazine*, IEEE, v. 10, n. 2, p. 18–29, 2015.

HUANG, G.-B.; ZHOU, H.; DING, X.; ZHANG, R. Extreme learning machine for regression and multiclass classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, IEEE, v. 42, n. 2, p. 513–529, 2012.

HUANG, G.-B.; ZHU, Q.-Y.; SIEW, C.-K. Extreme learning machine: theory and applications. *Neurocomputing*, Elsevier, v. 70, n. 1-3, p. 489–501, 2006.

HUANG, J.; YU, Z. L.; CAI, Z.; GU, Z.; CAI, Z.; GAO, W.; YU, S.; DU, Q. Extreme learning machine with multi-scale local receptive fields for texture classification. *Multidimensional Systems and Signal Processing*, Springer, v. 28, n. 3, p. 995–1011, 2017.

HUANG, Q.; XIA, C.; WU, C.; LI, S.; WANG, Y.; SONG, Y.; KUO, C.-C. J. Semantic segmentation with reverse attention. *arXiv preprint arXiv:1707.06426*, 2017.

JARRETT, K.; KAVUKCUOGLU, K.; LECUN, Y. et al. What is the best multi-stage architecture for object recognition? In: IEEE. *Computer Vision, 2009 IEEE 12th International Conference on*. [S.l.], 2009. p. 2146–2153.

JAYASUNDARA, V.; JAYASEKARA, S.; JAYASEKARA, H.; RAJASEGARAN, J.; SENEVIRATNE, S.; RODRIGO, R. Textcaps: Handwritten character recognition with very small datasets. In: IEEE. *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. [S.l.], 2019. p. 254–262.

JENKINSON, M.; PECHAUD, M.; SMITH, S. Bet2: Mr-based estimation of brain, skull and scalp surfaces. In: TORONTO, ON. *Eleventh annual meeting of the organization for human brain mapping*. [S.l.], 2005. v. 17, p. 167.

JESSON, A.; ARBEL, T. Hierarchical mrf and random forest segmentation of ms lesions and healthy tissues in brain mri. In: *The Longitudinal MS Lesion Segmentation Challenge*. [S.l.: s.n.], 2015.

JIA, Y.; SHELHAMER, E.; DONAHUE, J.; KARAYEV, S.; LONG, J.; GIRSHICK, R.; GUADARRAMA, S.; DARRELL, T. Caffe: Convolutional architecture for fast feature embedding. In: ACM. *Proceedings of the 22nd ACM international conference on Multimedia*. [S.l.], 2014. p. 675–678.

JIN, Y. Surrogate-assisted evolutionary computation: Recent advances and future challenges. *Swarm and Evolutionary Computation*, v. 1, n. 2, p. 61 – 70, 2011. ISSN 2210-6502. Disponível em: <http://www.sciencedirect.com/science/article/pii/S2210650211000198>.

JONES, J. P.; PALMER, L. A. An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex. *Journal of neurophysiology*, Am Physiological Soc, v. 58, n. 6, p. 1233–1258, 1987.

KARPATE, Y.; COMMOWICK, O.; BARILLOT, C. Robust detection of multiple sclerosis lesions from intensity-normalized multi-channel mri. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. *Medical Imaging 2015: Image Processing*. [S.l.], 2015. v. 9413, p. 941314.

KENNEDY, J.; EBERHART, R. Particle swarm optimization. In: *Neural Networks, 1995. Proceedings., IEEE International Conference on*. [S.l.: s.n.], 1995. v. 4, p. 1942–1948.

KHADEMI, A.; VENETSANOPOULOS, A.; MOODY, A. R. Generalized method for partial volume estimation and tissue segmentation in cerebral magnetic resonance images. *Journal of Medical Imaging*, International Society for Optics and Photonics, v. 1, n. 1, p. 014002, 2014.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2012. p. 1097–1105.

LECOEUR, J.; FERRÉ, J.-C.; BARILLOT, C. Optimized supervised segmentation of ms lesions from multispectral mris. In: *MICCAI workshop on Medical Image Analysis on Multiple Sclerosis (validation and methodological issues)*. [S.l.: s.n.], 2009.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *nature*, Nature Publishing Group, v. 521, n. 7553, p. 436, 2015.

LECUN, Y.; BOSER, B. E.; DENKER, J. S.; HENDERSON, D.; HOWARD, R. E.; HUBBARD, W. E.; JACKEL, L. D. Handwritten digit recognition with a back-propagation network. In: *Advances in neural information processing systems*. [S.l.: s.n.], 1990. p. 396–404.

LECUN, Y.; BOTTOU, L.; BENGIO, Y.; HAFFNER, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, IEEE, v. 86, n. 11, p. 2278–2324, 1998.

LECUN, Y.; JACKEL, L.; BOTTOU, L.; CORTES, C.; DENKER, J. S.; DRUCKER, H.; GUYON, I.; MULLER, U. A.; SACKINGER, E.; SIMARD, P. et al. Learning algorithms for classification: A comparison on handwritten digit recognition. *Neural networks: the statistical mechanics perspective*, World Scientific Singapore, v. 261, p. 276, 1995.

LIANG, M.; HU, X. Recurrent convolutional neural network for object recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2015. p. 3367–3375.

LIN, T.-Y.; ROYCHOWDHURY, A.; MAJI, S. Bilinear cnn models for fine-grained visual recognition. In: *Proceedings of the IEEE International Conference on Computer Vision*. [S.l.: s.n.], 2015. p. 1449–1457.

LIU, X.; GAO, C.; LI, P. A comparative analysis of support vector machines and extreme learning machines. *Neural Networks*, Elsevier, v. 33, p. 58–66, 2012.

LIU, X.; LIN, S.; FANG, J.; XU, Z. Is extreme learning machine feasible? a theoretical assessment (part i). *IEEE Transactions on Neural Networks and Learning Systems*, IEEE, v. 26, n. 1, p. 7–20, 2015.

LLADÓ, X.; OLIVER, A.; CABEZAS, M.; FREIXENET, J.; VILANOVA, J. C.; QUILES, A.; VALLS, L.; RAMIÓ-TORRENTÀ, L.; ROVIRA, A. Segmentation of multiple sclerosis lesions in brain mri: A review of automated approaches. *Information Sciences*, Elsevier, v. 186, n. 1, p. 164–185, 2012.

LONG, J.; SHELHAMER, E.; DARRELL, T. Fully convolutional networks for semantic segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2015. p. 3431–3440.

MA, B.; LI, X.; XIA, Y.; ZHANG, Y. Autonomous deep learning: A genetic dcnn designer for image classification. *Neurocomputing*, Elsevier, 2019.

MAHBOD, A.; WANG, C.; SMEDBY, O. Automatic multiple sclerosis lesion segmentation using hybrid artificial neural networks. *MSSEG Challenge Proceedings: Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure*, v. 29, 2016.

MAZZIOTTA, J.; TOGA, A.; EVANS, A.; FOX, P.; LANCASTER, J.; ZILLES, K.; WOODS, R.; PAUS, T.; SIMPSON, G.; PIKE, B. et al. A probabilistic atlas and reference system for the human brain: International consortium for brain mapping (icbm). *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, The Royal Society, v. 356, n. 1412, p. 1293–1322, 2001.

MCDONNELL, M. D.; VLADUSICH, T. Enhanced image classification with a fast-learning shallow convolutional neural network. In: IEEE. *Neural Networks (IJCNN), 2015 International Joint Conference on*. [S.l.], 2015. p. 1–7.

MCKINLEY, R.; GUNDERSEN, T.; WAGNER, F.; CHAN, A.; WIEST, R.; REYES, M. Nabla-net: a deep dag-like convolutional architecture for biomedical image segmentation: application to white-matter lesion segmentation in multiple sclerosis. *MSSEG Challenge Proceedings: Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure*, p. 37, 2016.

MUSCHELLI, J.; SWEENEY, E.; MARONGE, J.; CRAINICEANU, C. Prediction of ms lesions using random forests. *Proceedings of the 1st MICCAI Challenge on Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure-MICCAI-MSSEG*, p. 45–50, 2016.

PANG, S.; YANG, X. Deep convolutional extreme learning machine and its application in handwritten digit classification. *Computational intelligence and neuroscience*, Hindawi, v. 2016, 2016.

POLMAN, C. H.; REINGOLD, S. C.; BANWELL, B.; CLANET, M.; COHEN, J. A.; FILIPPI, M.; FUJIHARA, K.; HAVRDOVA, E.; HUTCHINSON, M.; KAPPOS, L. et al. Diagnostic criteria for multiple sclerosis: 2010 revisions to the mcdonald criteria. *Annals of neurology*, Wiley Online Library, v. 69, n. 2, p. 292–302, 2011.

RAHIMI, A.; RECHT, B. Random features for large-scale kernel machines. In: CURRAN ASSOCIATES INC. *Proceedings of the 20th International Conference on Neural Information Processing Systems*. [S.l.], 2007. p. 1177–1184.

RAHIMI, A.; RECHT, B. Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2009. p. 1313–1320.

RAVISHANKAR, H.; SUDHAKAR, P.; VENKATARAMANI, R.; THIRUVENKADAM, S.; ANNANGI, P.; BABU, N.; VAIDYA, V. Understanding the mechanisms of deep transfer learning for medical images. In: *Deep Learning and Data Labeling for Medical Applications*. [S.l.]: Springer, 2016. p. 188–196.

RAZAVIAN, A. S.; AZIZPOUR, H.; SULLIVAN, J.; CARLSSON, S. Cnn features off-the-shelf: an astounding baseline for recognition. In: IEEE. *Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on*. [S.l.], 2014. p. 512–519.

REN, S.; HE, K.; GIRSHICK, R.; SUN, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2015. p. 91–99.

ROURA, E.; OLIVER, A.; CABEZAS, M.; VALVERDE, S.; PARETO, D.; VILANOVA, J. C.; RAMIÓ-TORRENTÀ, L.; ROVIRA, À.; LLADÓ, X. A toolbox for multiple sclerosis lesion segmentation. *Neuroradiology*, Springer, p. 1–13, 2015.

RUDI, A.; ROSASCO, L. Generalization properties of learning with random features. In: *Advances in Neural Information Processing Systems*. [S.l.: s.n.], 2017. p. 3218–3228.

RUMELHART, D.; HINTON, G.; WILLIAMS, R. Learning internal representations by error propagation. In: MIT PRESS. *Parallel distributed processing: explorations in the microstructure of cognition, vol. 1*. [S.l.], 1986. p. 318–362.

RUSS, T. D.; KOCH, M. W.; LITTLE, C. Q. A 2d range hausdorff approach for 3d face recognition. In: IEEE. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)-Workshops*. [S.l.], 2005. p. 169–169.

SANTOS, M. M.; DINIZ, P.; SILVA-FILHO, A. G.; SANTOS, W. P. Evaluation-oriented training strategy on ms segmentation challenge 2016. In: COMMOWICK, O.; CERVENANSKY, F.; AMELI, R. (Ed.). *Proceedings of the 1st MICCAI Challenge on Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure–MICCAI-MSSEG*. [S.l.: s.n.], 2016. p. 57–62.

SANTOS, M. M.; DINIZ, P. R. B.; SILVA-FILHO, A. G.; SANTOS, W. P. Evaluation-oriented training via surrogate metrics for multiple sclerosis segmentation. In: OURSELIN, S.; JOSKOWICZ, L.; SABUNCU, M. R.; UNAL, G.; WELLS, W. (Ed.). *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*. Cham: Springer International Publishing, 2016. p. 398–405. ISBN 978-3-319-46723-8.

SANTOS, M. M. dos; FILHO, A. G. da S.; SANTOS, W. P. dos. Deep convolutional extreme learning machines: Filters combination and error model validation. *Neurocomputing*, Elsevier, v. 329, p. 359–369, 2019.

SATO, I.; NISHIMURA, H.; YOKOI, K. Apac: Augmented pattern classification with neural networks. *arXiv preprint arXiv:1505.03229*, 2015.

SAXE, A. M.; KOH, P. W.; CHEN, Z.; BHAND, M.; SURESH, B.; NG, A. Y. On random weights and unsupervised feature learning. In: OMNIPRESS. *Proceedings of the 28th International Conference on International Conference on Machine Learning*. [S.l.], 2011. p. 1089–1096.

SCHERER, D.; MÜLLER, A.; BEHNKE, S. Evaluation of pooling operations in convolutional architectures for object recognition. In: SPRINGER. *International conference on artificial neural networks*. [S.l.], 2010. p. 92–101.

SCHLEGL, T.; WALDSTEIN, S. M.; VOGL, W.-D.; SCHMIDT-ERFURTH, U.; LANGS, G. Predicting semantic descriptions from medical images with convolutional neural networks. In: SPRINGER. *International Conference on Information Processing in Medical Imaging*. [S.l.], 2015. p. 437–448.

SCHWARZ, M.; SCHULZ, H.; BEHNKE, S. Rgb-d object recognition and pose estimation based on pre-trained convolutional neural network features. In: IEEE. *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. [S.l.], 2015. p. 1329–1335.

SERMANET, P.; CHINTALA, S.; LECUN, Y. Convolutional neural networks applied to house numbers digit classification. In: IEEE. *Pattern Recognition (ICPR), 2012 21st International Conference on*. [S.l.], 2012. p. 3288–3291.

SHAH, M.; XIAO, Y.; SUBBANNA, N.; FRANCIS, S.; ARNOLD, D. L.; COLLINS, D. L.; ARBEL, T. Evaluating intensity normalization on mris of human brain with multiple sclerosis. *Medical image analysis*, Elsevier, v. 15, n. 2, p. 267–282, 2011.

SHAWON, A.; RAHMAN, M. J.-U.; MAHMUD, F.; ZAMAN, M. A. Bangla handwritten digit recognition using deep cnn for large and unbiased dataset. In: IEEE. *2018 International Conference on Bangla Speech and Language Processing (ICBSLP)*. [S.l.], 2018. p. 1–6.

SHIN, H.-C.; ROTH, H. R.; GAO, M.; LU, L.; XU, Z.; NOGUES, I.; YAO, J.; MOLLURA, D.; SUMMERS, R. M. Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning. *IEEE transactions on medical imaging*, IEEE, v. 35, n. 5, p. 1285–1298, 2016.

SINGH, S.; PAUL, A.; ARUN, M. Parallelization of digit recognition system using deep convolutional neural network on cuda. In: IEEE. *Sensing, Signal Processing and Security (ICSSS), 2017 Third International Conference on*. [S.l.], 2017. p. 379–383.

SLED, J. G.; ZIJDENBOS, A. P.; EVANS, A. C. A nonparametric method for automatic correction of intensity nonuniformity in mri data. *Medical Imaging, IEEE Transactions on*, IEEE, v. 17, n. 1, p. 87–97, 1998.

SMITH, S. M. Fast robust automated brain extraction. *Human brain mapping*, Wiley Online Library, v. 17, n. 3, p. 143–155, 2002.

SOUPLET, J.-C.; LEBRUN, C.; AYACHE, N.; MALANDAIN, G. et al. An automatic segmentation of t2-flair multiple sclerosis lesions. In: *The MIDAS Journal-MS Lesion Segmentation (MICCAI 2008 Workshop)*. [S.l.: s.n.], 2008.

STYNER, M.; LEE, J.; CHIN, B.; CHIN, M. S.; TRAN, H. huong; JEWELLS, V.; WARFIELD, S. 3d segmentation in the clinic: A grand challenge ii: Ms lesion segmentation. In: *MICCAI 2008 Workshop*. [S.l.: s.n.], 2008. p. 1–5.

TAHA, A. A.; HANBURY, A. Metrics for evaluating 3d medical image segmentation: analysis, selection, and tool. *BMC medical imaging*, BioMed Central Ltd, v. 15, n. 1, p. 29, 2015.

THOMPSON, A. J.; BANWELL, B. L.; BARKHOF, F.; CARROLL, W. M.; COETZEE, T.; COMI, G.; CORREALE, J.; FAZEKAS, F.; FILIPPI, M.; FREEDMAN, M. S. et al. Diagnosis of multiple sclerosis: 2017 revisions of the mcdonald criteria. *The Lancet Neurology*, Elsevier, v. 17, n. 2, p. 162–173, 2018.

TOMAS-FERNANDEZ, X.; WARFIELD, S. A model of population and subject (mops) intensities with application to multiple sclerosis lesion segmentation. *Medical Imaging, IEEE Transactions on*, v. 34, n. 6, p. 1349–1361, June 2015. ISSN 0278-0062.

TUSTISON, N. J.; AVANTS, B. B.; COOK, P. A.; ZHENG, Y.; EGAN, A.; YUSHKEVICH, P. A.; GEE, J. C. N4itk: improved n3 bias correction. *IEEE transactions on medical imaging*, NIH Public Access, v. 29, n. 6, p. 1310, 2010.

URIEN, H.; BUVAT, I.; ROUGON, N.; SOUSSAN, M.; BLOCH, I. Brain lesion detection in 3d pet images using max-trees and a new spatial context criterion. In: SPRINGER. *International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing*. [S.l.], 2017. p. 455–466.

VAIDYA, S.; CHUNDURU, A.; MUTHUGANAPATHY, R.; KRISHNAMURTHI, G. Longitudinal multiple sclerosis lesion segmentation using 3d convolutional neural networks. In: *The Longitudinal MS Lesion Segmentation Challenge*. [S.l.: s.n.], 2015.

VALVERDE, S.; CABEZAS, M.; ROURA, E.; GONZÁLEZ-VILLÀ, S.; PARETO, D.; VILANOVA, J. C.; RAMIÓ-TORRENTÀ, L.; ROVIRA, À.; OLIVER, A.; LLADÓ, X. Improving automated multiple sclerosis lesion segmentation with a cascaded 3d convolutional neural network approach. *NeuroImage*, Elsevier, v. 155, p. 159–168, 2017.

VAPNIK, V. *The nature of statistical learning theory*. [S.l.]: Springer-Verlag New York, 2000.

VERA-OLMOS, F.; MELERO, H.; MALPICA, N. Random forest for multiple sclerosis lesion segmentation. *MSSEG Challenge Proceedings: Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure*, p. 81, 2016.

VOS, T.; BARBER, R. M.; BELL, B.; BERTOZZI-VILLA, A.; BIRYUKOV, S.; BOLLIGER, I.; CHARLSON, F.; DAVIS, A.; DEGENHARDT, L.; DICKER, D. et al. Global, regional, and national incidence, prevalence, and years lived with disability for 301 acute and chronic diseases and injuries in 188 countries, 1990–2013: a systematic analysis for the global burden of disease study 2013. *The Lancet*, Elsevier, v. 386, n. 9995, p. 743–800, 2015.

WAN, L.; ZEILER, M.; ZHANG, S.; CUN, Y. L.; FERGUS, R. Regularization of neural networks using dropconnect. In: *International Conference on Machine Learning*. [S.l.: s.n.], 2013. p. 1058–1066.

WARFIELD, S. K.; ZOU, K. H.; WELLS, W. M. Simultaneous truth and performance level estimation (staple): an algorithm for the validation of image segmentation. *IEEE transactions on medical imaging*, NIH Public Access, v. 23, n. 7, p. 903, 2004.

YANG, F.; CHOI, W.; LIN, Y. Exploit all the layers: Fast and accurate cnn object detector with scale dependent pooling and cascaded rejection classifiers. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2016. p. 2129–2137.

YOUSEFI-AZAR, M.; MCDONNELL, M. D. Semi-supervised convolutional extreme learning machine. In: IEEE. *Neural Networks (IJCNN), 2017 International Joint Conference on*. [S.l.], 2017. p. 1968–1974.

ZHAN, T.; ZHAN, Y.; LIU, Z.; XIAO, L.; WEI, Z. Automatic method for white matter lesion segmentation based on t1-fluid-attenuated inversion recovery images. *IET Computer Vision*, IET, 2015.

ZHANG, C.; BENGIO, S.; HARDT, M.; RECHT, B.; VINYALS, O. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

ZHOU, B.; LAPEDRIZA, A.; XIAO, J.; TORRALBA, A.; OLIVA, A. Learning deep features for scene recognition using places database. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2014. p. 487–495.

ZHU, W.; MIAO, J.; QING, L. Constrained extreme learning machine: a novel highly discriminative random feedforward neural network. In: IEEE. *Neural Networks (IJCNN), 2014 International Joint Conference on*. [S.l.], 2014. p. 800–807.

ZIJDENBOS, A. P.; FORGHANI, R.; EVANS, A. C. Automatic "pipeline" analysis of 3-d mri data for clinical trials: Application to multiple sclerosis. *Medical Imaging, IEEE Transactions on*, IEEE, v. 21, n. 10, p. 1280–1291, 2002.

APÊNDICE A – ARTIGO NO CONGRESSO MICCAI2016

Reprodução do Registro Bibliográfico para Identificação

SANTOS, M. M.; DINIZ, P. R. B.; SILVA-FILHO, A. G.; SANTOS, W. P. Evaluation-oriented training via surrogate metrics for multiple sclerosis segmentation. In: OURSELIN, S.; JOSKOWICZ, L.; SABUNCU, M. R.; UNAL, G.; WELLS, W. (Ed.). *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*. Cham: Springer International Publishing, 2016. p. 398–405. ISBN 978-3-319-46723-8.

Nota sobre Direitos Autorais

A versão final do artigo anterior à publicação é reproduzida a seguir com permissão da Springer Nature: Springer International Publishing AG, Medical Image Computing e Computer-Assisted Intervention - MICCAI 2016 por Ourselin S., Joskowicz L., Sabuncu M., Unal G., Wells W. (eds) © (2016). A versão publicada está disponível em <https://doi.org/10.1007/978-3-319-46723-8_46>.

Copyright Notice

The final version of the article previous to publication is reprinted in the following by permission from Springer Nature: Springer International Publishing AG, Medical Image Computing and Computer-Assisted Intervention - MICCAI 2016 by Ourselin S., Joskowicz L., Sabuncu M., Unal G., Wells W. (eds) © (2016). The published version is available at <https://doi.org/10.1007/978-3-319-46723-8_46>.

Evaluation-Oriented Training via Surrogate Metrics for Multiple Sclerosis Segmentation

Michel M. Santos[1], Paula R. B. Diniz[2], Abel G. Silva-Filho[1], and Wellington P. Santos[3],

[1] Centro de Informática, Universidade Federal de Pernambuco, Pernambuco, Brazil
[2] Núcleo de Telessaúde, Universidade Federal de Pernambuco, Pernambuco, Brazil
[3] Dpto. de Eng. Biomédica, Universidade Federal de Pernambuco, Pernambuco, Brazil
michelmozinho@gmail.com

Abstract. In current approaches to automatic segmentation of multiple sclerosis (MS) lesions, the segmentation model is not optimized with respect to all relevant evaluation metrics at once, leading to unspecific training. An obstacle is that the computation of relevant metrics is three-dimensional (3D). The high computational costs of 3D metrics make their use impractical as learning targets for iterative training. In this paper, we propose an oriented training strategy that employs cheap 2D metrics as surrogates for expensive 3D metrics. We optimize a simple multilayer perceptron (MLP) network as segmentation model. We study fidelity and efficiency of surrogate 2D metrics. We compare oriented training to unspecific training. The results show that oriented training produces a better balance between metrics surpassing unspecific training on average. The segmentation quality obtained with a simple MLP through oriented training is comparable to the state-of-the-art; this includes a recent work using a deep neural network, a more complex model. By optimizing all relevant evaluation metrics at once, oriented training can improve MS lesion segmentation.

1 Introduction

In most multiple sclerosis (MS) lesion segmentation methods specific metrics serve as evaluation criteria, but not as training targets. For instance, many studies use dice similarity coefficient, sensitivity, specificity, and surface distance as evaluation metrics, but have energy, likelihood, or maximum *a posteriori* functions as the learning target [1–4]. Despite previous efforts, automatic MS segmentation still requires improvements [5, 6]. The use of evaluation metrics as optimization targets has potential to improve MS segmentation methods [7]. Evaluation metrics of clinical relevance involve lesion count, volume, and shape that are criteria to characterize the MS progression [8]. However, previous works failed to target all metrics of clinical relevance at the same time [7, 9, 10].

An obstacle to evaluation-oriented training is that domain-specific metrics can be too expensive for iterative optimization of MS segmentation methods. The high computational cost is in part due to the three-dimensional (3D) computation of evaluation metrics. For instance, a shape based metric, as the average

surface distance, may take minutes to be computed in three dimensions [11]. In addition, the measures based on counts of overlapped regions between segmentation and ground truth depend in turn on connected components method to obtain contiguous regions. The 3D computation of connected components can take minutes as well [12]. To verify that the optimization of 3D metrics is impractical, suppose that the computation of measures spends 10 minutes for each one of 10 patients. In this case, considering 1000 iterations of an optimization algorithm, the training would take more than two months (100,000 minutes). As an alternative, surrogate-assisted optimization, a field of evolutionary computing, uses efficient computational models to approximate costly target functions to be optimized [13]. While surrogate-assisted optimization has been applied to other fields, no studies have been found in MS lesion segmentation. The surrogate-assisted method can make oriented training feasible enabling the optimization of a segmentation model specifically for MS lesion detection.

The contributions of this paper are as follows: a surrogated-assisted training method that allows indirect optimization of costly and multiple metrics, specially designed for MS segmentation; the method comprises a surrogate objective function; we apply the proposed optimization method to carry out evaluation-oriented training, aiming metrics of lesion count, volume, and shape, all at once.

2 MS Segmentation Challenge: Dataset and 3D Metrics

The MS Lesion Segmentation Challenge 2008 aims to compare automatic methods [14]. The competition website[4] is still providing image data and receiving new submissions. For each patient in the database, three images were acquired with different MRI sequences called T1-weighted (T1w), T2-weighted (T2w) and fluid-attenuated inversion recovery (FLAIR). All images were coregistered and resampled to a matrix of 512x512x512, corresponding to a voxel size of 0.5x0.5x0.5mm. The segmentation was made by human experts, attributing to each voxel the values: 1, for lesion; 0, otherwise. In the competition, the images are divided into two sets: one named *training* and other named *test*. To avoid confusion with the stages of cross-validation, in our study, we renamed training and test sets from competition to *public* and *private*, respectively, according to the availability of ground truth. For each submission, the competition website attributes a score based on the segmentation done over the private set. This evaluation score comprises four metrics that are 3D computed: relative absolute volume difference (RAVD); average symmetric surface distance (ASSD); lesion-wise true positive rate (LTPR); lesion-wise false positive rate (LFPR). The final (3D) score is an average scaled to be between 0 and 100, the higher, the better. A score of 90 is equivalent to the expected performance of an independent human expert. The original evaluation algorithm is available online[5].

[4] Available at http://www.ia.unc.edu/MSseg/
[5] https://github.com/telnet2/gradworks/tree/master/EvalSegmentation

3 Segmentation Model and Optimization Method

In our approach, we adopt a simple multilayer perceptron (MLP) model with a single hidden layer containing just a few neurons, to accelerate the algorithm execution and enable statistical evaluation. This shallow neural network serves to demonstrate the impact of oriented training and should not be disregarded. Shallow neural networks, when properly parameterized and trained, can achieve performance equivalent to that of state-of-the-art deep learning models [15].

We employ particle swarm optimization (PSO) for training MLP as an alternative to backpropagation [16]. The backpropagation algorithm restricts the form of learning target. On the other hand, PSO enables to use different learning targets without being necessary to develop a specific procedure for each function to be optimized. In our implementation, the MLP had three input neurons that correspond to the intensity values for T1w, T2w and FLAIR images at each voxel position, two neurons at output layer that indicate the voxel class, and a single hidden layer with six neurons and logistic activation function. In PSO, 20 particles were used, besides the parameters $w = 0.65$, $c_1 = 3$, $c_2 = 1$. The search space at each coordinate was limited to the range [-20,20]. Maximum velocity was set to 20% of search space width. When some particle went to outside the search space, its velocity was inverted and reduced by a factor of -0.01. All parameters were defined empirically or gathered on literature.

4 Evaluation-Oriented Training through Surrogate Metrics

4.1 Computationaly Cheaper Training Targets

The 3D computation of learning targets would be unfeasible for iterative training. To reduce the computational burden, instead of costly 3D metrics, we employed cheaper 2D surrogates. Accordingly, the surrogate 2D-score depends on four 2D metrics as follows: ρ_1: *relative absolute area difference*; ρ_2: *average symmetric surface distance*; ρ_3: *lesion-wise true positive rate* ; ρ_4: *lesion-wise false positive rate*. As in the 2008 competition (Section 2), the four metrics are transformed to the range [0, 100] with a 90 score being equivalent to the performance of a human expert. Thus, the learning target for evaluation-oriented training is the 2D-Score $= (\rho_1 + \rho_2 + \rho_3 + \rho_4)/4$. The optimization of all the four metrics is important: a single metric that reaches just a 50% score represents a loss of 12.5 points on the final score, which is too much in such a competitive task. The 2D-score is faster because it can be calculated in lower dimension and averaged over a reduced number of selected slices (as we will see). Furthermore, notice that surrogate metrics are faster due to problem reduction, regardless of hardware-based acceleration.

4.2 Evaluation-Oriented Versus Unspecific Training

The proposed evaluation-oriented training differs from unspecific training in two aspects: learning target and data sampling. In evaluation-oriented training, the

objective function (Algorithm 1) has as learning target the 2D-score. In contrast, the unspecific training has the MSE as learning target in the objective function (Algorithm 2). Regarding data sampling, the oriented objective computes an average of four 2D metrics that requires samples of slices for each patient. On the other hand, samples of voxels are used with unspecific training for MSE calculation. Both oriented and unspecific objectives include a pre-classification rule. This rule prevents unnecessary computation of MLP outputs and takes advantage of the fact that lesions appear hyper-intense in FLAIR images (Algorithm 1 at line 5 and Algorithm 2 at line 4). Notice that such rule requires standardized intensities. We standardized images to have median 0 and interquartile range 1. In unspecific training, to balance classes, for each patient, all lesion voxels are taken, and an equal number of non-lesion voxels is sampled. Note that 2D-score must be maximized, whereas MSE must be minimized. For each objective function, the PSO was configured and employed accordingly.

Algorithm 1 Oriented Objective

```
1: function OBJECTIVE(sliceSamples)
2:     for each patient do
3:         for each slice in sliceSamples do
4:             for each voxel do
5:                 if FLAIR intensity < 0 then
6:                     assign non-lesion to voxel
7:                 else
8:                     classify voxel via MLP
9:                 end if
10:            end for
11:            compute 2D (per slice) metrics
12:        end for
13:    end for
14:    compute average 2D metrics
15:    objective = 2D-score          ▷ Section (4.1)
16: end function
```

Algorithm 2 Unspecific Objective

```
1: function OBJECTIVE(voxelSamples)
2:     for each patient do
3:         for each voxel in voxelSamples do
4:             if FLAIR intensity < 0 then
5:                 assign non-lesion to voxel
6:             else
7:                 classify voxel via MLP
8:             end if
9:         end for
10:        compute error per voxel
11:    end for
12:    objective = MSE
13: end function
```

5 Results and Discussion

5.1 Experimental Setup

Fidelity and Efficiency of 2D-Score. The 2D-score was evaluated regarding fidelity and efficiency as a function of the number of slices. The plane, quantity, and position of the selected slices can influence the final 2D-score. We selected slices in the axial plane. The number of slices s was varied as follows: $s = \{2^2, 2^3, \ldots, 2^9\}$. Position p_n of the n-th slice was given by $p_n = 1 + (n-1)(2^9/s)$ when s slices were selected. For example, for $s = 4$, the selected slices were $\{1, 129, 257, 385\}$. In addition, we evaluated a case with three slices in the positions 240, 250 and 260 by being approximately medial slices. A 2D-score based on s slices per patient is denoted as 2D-score-s.

In this analysis, it was employed a subset of 10 images that was segmented by two human experts, totaling 20 segmentations. For each of these segmentations, 2D-score and 3D-score were computed as measures of rater-rater agreement. Fidelity was given by the (Pearson) correlation of the 2D-score with the 3D-score. Efficiency was assessed via the median time taken per patient to compute

each 2D-score. Note that this analysis is independent of automatic methods, since we employ only segmentations made by human raters.

Comparison Based on Public and Private Data. The images with public ground truth serve for statistical evaluation of training methods since they can be used to compute segmentation metrics locally and repeatedly. We employed a repeated holdout method to compare the training algorithms statistically. For each type of training, the execution of the PSO-MLP was repeated 120 times. We compare two training algorithms: oriented using 2D-score-3; unspecific using MSE. We computed the median execution time and the median number of objective function evaluations to assess training efficiency.

The purpose of the private ground truth data is to avoid the optimization of algorithms to a particular set of images. We compare the oriented training with the unspecific training on private data as well. For this, we selected the MLP with the best learning target regarding all images in the public dataset. Next, each selected MLP was applied to detect lesions on images with a private ground truth. The training methods are compared through the 3D-score provided by the competition website. We also compare oriented training to state-of-the-art methods. For this, we employed a Gaussian filter as a post-processing step to avoid false-positives due to small candidate lesions, like other approaches [2, 9].

5.2 Surrogate 2D Versus Actual 3D Metrics

Fidelity and efficiency of 2D-score as a surrogate for the 3D-score are examined. The fidelity tends to grow with the number of slices used to calculate the 2D-score (Table 1). On the other hand, the efficiency of 2D-score decreases with more slices. Note that the 2D-score has positive fidelity even with a small number of slices. A positive correlation, even weak, indicates a tendency for 3D-score increases as 2D-score increases. This monotonic relationship makes reasonable the use of the low-fidelity 2D-score-3 as objective function. Moreover, the low-fidelity 2D-score-3 is around 80 times faster than the high-fidelity 2D-score-512.

Table 1: Fidelity and efficiency of 2D-score as surrogate for 3D-score

	Number of slices in 2D-score								
	3	4	8	16	32	64	128	256	512
Correlation with 3D-score	0.13	0.15	0.17	0.19	0.28	0.29	0.49	0.77	0.81
Median time per patient (s)	0.04	0.04	0.06	0.11	0.22	0.44	0.88	1.73	3.46

5.3 Training Efficiency Using Public Ground Truth

We compare oriented to the unspecific training with respect to the number of function evaluations and execution time (Table 2). The evaluation-oriented train-

ing stopped with lower objective function evaluations but took more time overall ($\sim$ 1 hour) compared to unspecific training ($\sim$ 1 minute), because MSE is computed faster than surrogate measures. On the other hand, compared to 3D metrics, the 2D surrogates can reduce the overall training time from months to hours. In oriented training, the time per evaluation of the objective function is roughly $3238/1645 = 1.9s$. With 10 individuals in the training set, the time per patient in a single oriented objective computation is $0.19s$ (note that the time for obtaining just the 2D-score-3 is $0.04s$ per patient). Once the training is concluded, the time taken to detect lesions in a new patient is about 40 seconds.

Table 2: Training method efficiency

	Unspecific	Oriented
Median objective evaluations	2911	1645
Median execution time (s)	78.81	3238.48

5.4 Segmentation Effectiveness on Private Ground Truth

Evaluation-oriented training with 2D-score-3 surpasses unspecific training in segmentation quality as given by the average 3D-score (Table 3, best scores in bold). Unspecific training with MSE has better performance only in the true-positive score. These results reveal in which metrics the oriented training has the advantage. Furthermore, the proposed oriented training with 2D-score-3 better balances the four metrics and surpasses the unspecific training in the average score. A low-fidelity 2D-score with just 3 slices enables evaluation-oriented training to reach a better segmentation quality than unspecific training.

Table 3: Segmentation quality per learning target on private data

Target	RAVD		ASSD		LTPR		LFPR		Average 3D-Score
	[%]	Score	[mm]	Score	[%]	Score	[%]	Score	
MSE	2302	6	14	71	88	**98**	97	51	56.30
2D-score-3	63	**91**	6	**87**	43	76	49	**80**	**83.26**

5.5 Comparison with State-of-the-Art Methods

Compared with state-of-the-art methods, the evaluation-oriented training is the first to optimize measures of lesion count, volume, and shape through surrogate learning targets (Table 4). The proposed method reaches a segmentation quality

score comparable to that of state-of-the-art methods surpassing some recent approaches. Interestingly, our oriented method using a simple and shallow neural network model reaches a 3D-score of 83.26 that is very close to 84.07 obtained by a state-of-the-art deep neural network [17].

Table 4: Comparison with state-of-the-art methods

Optimized Targets*	3D-Score	Reference
Maximum a Posteriori	86.93	[4]
Rotation-invariant Similarity	86.10	[18]
Trimmed Likelihood, Energy Function	84.46	[3]
MSE Weighted by Sensitivity	84.07	[17]
$\Rightarrow$ LTPR, LFPR, RAVD, ASSD	83.26	present work
Energy Function, ROC, Jaccard	82.54	[10]
Dice (voxel-wise), LTPR, LPPV	82.34	[9]
Log-likelihood	79.99	[1]
Trimmed Likelihood	79.09	[2]

* lesion-wise positive predictive value (LPPV) and receiver operating characteristic (ROC)

6 Conclusion

In this paper, we introduced an evaluation-oriented training strategy that targets evaluation criteria of MS lesion segmentation. To make the evaluation-oriented training feasible, we optimize cheaper 2D surrogates rather than costly 3D metrics. Different from previous methods, the evaluation-oriented training targets metrics of lesion count, volume, and shape, all at once. We found that oriented training significantly improves the MS segmentation quality compared to unspecific training. Moreover, the oriented training of a simple MLP yielded a segmentation quality that is comparable to the state-of-the-art performance. Surrogate metrics are a promising approach to optimize MS segmentation methods. Future studies should investigate oriented training for other segmentation models, evaluate slice selection strategies, and aim the 2D-score using more slices.

References

1. Souplet, J.C., Lebrun, C., Ayache, N., Malandain, G., et al.: An automatic segmentation of T2-FLAIR multiple sclerosis lesions. In: The MIDAS Journal-MS Lesion Segmentation (MICCAI 2008 Workshop) (2008)
2. García-Lorenzo, D., Prima, S., Arnold, D.L., Collins, D.L., Barillot, C.: Trimmed-likelihood estimation for focal lesions and tissue segmentation in multisequence MRI for multiple sclerosis. Medical Imaging, IEEE Transactions on 30(8), 1455–1467 (2011)

3. Tomas-Fernandez, X., Warfield, S.: A model of population and subject (MOPS) intensities with application to multiple sclerosis lesion segmentation. Medical Imaging, IEEE Transactions on 34(6), 1349–1361 (June 2015)
4. Jesson, A., Arbel, T.: Hierarchical MRF and random forest segmentation of MS lesions and healthy tissues in brain MRI. In: The Longitudinal MS Lesion Segmentation Challenge (2015)
5. Lladó, X., Oliver, A., Cabezas, M., Freixenet, J., Vilanova, J.C., Quiles, A., Valls, L., Ramió-Torrentà, L., Rovira, A.: Segmentation of multiple sclerosis lesions in brain MRI: A review of automated approaches. Information Sciences 186(1), 164–185 (2012)
6. García-Lorenzo, D., Francis, S., Narayanan, S., Arnold, D.L., Collins, D.L.: Review of automatic segmentation methods of multiple sclerosis white matter lesions on conventional magnetic resonance imaging. Medical image analysis 17(1), 1–18 (2013)
7. Lecoeur, J., Ferré, J.C., Barillot, C.: Optimized supervised segmentation of MS lesions from multispectral MRIs. In: MICCAI workshop on Medical Image Analysis on Multiple Sclerosis (validation and methodological issues) (2009)
8. Barkhof, F., Filippi, M., Miller, D.H., Scheltens, P., Campi, A., Polman, C.H., Comi, G., Ader, H.J., Losseff, N., Valk, J.: Comparison of MRI criteria at first presentation to predict conversion to clinically definite multiple sclerosis. Brain 120(11), 2059–2069 (1997)
9. Roura, E., Oliver, A., Cabezas, M., Valverde, S., Pareto, D., Vilanova, J.C., Ramió-Torrentà, L., Rovira, À., Lladó, X.: A toolbox for multiple sclerosis lesion segmentation. Neuroradiology pp. 1–13 (2015)
10. Zhan, T., Zhan, Y., Liu, Z., Xiao, L., Wei, Z.: Automatic method for white matter lesion segmentation based on T1-fluid-attenuated inversion recovery images. IET Computer Vision (2015)
11. Taha, A.A., Hanbury, A.: Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool. BMC medical imaging 15(1), 29 (2015)
12. He, L., Chao, Y., Suzuki, K.: Two efficient label-equivalence-based connected-component labeling algorithms for 3-D binary images. Image Processing, IEEE Transactions on 20(8), 2122–2134 (2011)
13. Jin, Y.: Surrogate-assisted evolutionary computation: Recent advances and future challenges. Swarm and Evolutionary Computation 1(2), 61–70 (2011)
14. Styner, M., Lee, J., Chin, B., Chin, M.S., huong Tran, H., Jewells, V., Warfield, S.: 3D segmentation in the clinic: A grand challenge II: MS lesion segmentation. In: MICCAI 2008 Workshop. pp. 1–5 (2008)
15. Ba, J., Caruana, R.: Do deep nets really need to be deep? In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 27, pp. 2654–2662. Curran Associates, Inc. (2014)
16. Eberhart, R.C., Shi, Y.: Computational Intelligence - Concepts to Implementations. Elsevier (2007)
17. Brosch, T., Tang, L.Y.W., Yoo, Y., Li, D.K.B., Traboulsee, A., Tam, R.: Deep 3D convolutional encoder networks with shortcuts for multiscale feature integration applied to multiple sclerosis lesion segmentation. Medical Imaging, IEEE Transactions on 35(5), 1229–1239 (May 2016)
18. Guizard, N., Coupé, P., Fonov, V.S., Manjón, J.V., Arnold, D.L., Collins, D.L.: Rotation-invariant multi-contrast non-local means for MS lesion segmentation. NeuroImage: Clinical 8(0), 376 – 389 (2015)

APÊNDICE B – ARTIGO NO *WORKSHOP* MSSEG2016

Reprodução do Registro Bibliográfico para Identificação

SANTOS, M. M.; DINIZ, P.; SILVA-FILHO, A. G.; SANTOS, W. P. Evaluation-oriented training strategy on ms segmentation challenge 2016. In: COMMOWICK, O.; CERVENANSKY, F.; AMELI, R. (Ed.). *Proceedings of the 1st MICCAI Challenge on Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure–MICCAI-MSSEG*. [S.l.: s.n.], 2016. p. 57–62.

Nota sobre Direitos Autorais

O artigo consta nos anais da competição de segmentação de lesões de esclerose múltipla do MICCAI2016, publicado em repositório institucional de acesso aberto no endereço <https://www.hal.inserm.fr/inserm-01397806> e reproduzido a seguir. Os direitos sobre o artigo são mantidos plenamente pelos autores.

Copyright Notice

The article is in the proceedings of Multiple Sclerosis Lesion Segmentation Challenge of MICCAI2016, published in an open access institutional repository at <https://www.hal.inserm.fr/inserm-01397806> and reproduced below. The rights to the article are fully retained by the authors.

Evaluation-Oriented Training Strategy on MS Segmentation Challenge 2016

Michel M. Santos[1], Paula R. B. Diniz[2], Abel G. Silva-Filho[1], and Wellington P. Santos[3,]

[1] Centro de Informática, Universidade Federal de Pernambuco, Pernambuco, Brazil
[2] Núcleo de Telessaúde, Universidade Federal de Pernambuco, Pernambuco, Brazil
[3] Dpto. de Eng. Biomédica, Universidade Federal de Pernambuco, Pernambuco, Brazil
michelmozinho@gmail.com

Abstract. In most of the current approaches to automatic segmentation of multiple sclerosis (MS) lesions, the segmentation methods are not optimized with respect to all relevant evaluation metrics at once. An obstacle is that the computation of relevant metrics is three-dimensional (3D). The high computational costs of 3D metrics make their use impractical as learning targets for iterative training. In a companion paper, we propose an oriented training strategy that targets cheap 2D metrics as surrogates for costly 3D metrics. We applied the evaluation-oriented training with surrogate learning targets to optimize a simple multilayer perceptron (MLP) network at the core of a segmentation pipeline. In a previous competition, our segmentation strategy achieved a performance that was comparable to state-of-the-art methods. In this paper, by the opportunity of the MS Segmentation Challenge 2016, we apply the proposed method on a larger and more diverse dataset. We expect to compare the proposed strategy to other methods concerning segmentation quality and runtime on the computational cloud provided by the challenge organizers.

1 Introduction

In most methods for multiple sclerosis (MS) lesion segmentation, specific metrics serve as evaluation criteria, but not as optimization targets. Usual evaluation metrics involve lesion count, volume, and shape [1], representing clinical criteria of disease progression [2]. On the other hand, many methods have energy, likelihood, maximum *a posteriori*, or squared error as the optimization target [3–6].

An obstacle to evaluation-oriented training is that evaluation metrics may be too costly for iterative training. The high computational cost is in part due to the three-dimensional computation of evaluation metrics. A shape-based metric, as the average surface distance, may take seconds to be computed in three dimensions [7]. Moreover, the measures based on counts of overlapping lesions between the predicted segmentation and ground truth depend on obtaining contiguous regions. To obtain contiguous lesions, the connected components method

can take seconds in 3D images as well [8]. Suppose for instance that the computation of measures spends 1 minute for each one of 10 patients. In this case, for 1000 iterations of an optimization algorithm, a single training would take almost a week (10,000 minutes). In a companion paper [9], we propose cheaper 2D metrics as surrogates for costly 3D metrics to make the oriented training feasible.

In this paper, we apply the proposed strategy to images of the MS Segmentation Challenge 2016. Compared to a previous challenge [1], the current competition is an opportunity to assess our pipeline on a larger data set involving more sequences, expert segmentations, and scanners.

2 Material and Methods

2.1 MS Segmentation Challenge 2016

Data Set and Preprocessing The MS Lesion Segmentation Challenge 2016 aims to assess automatic methods on a common infrastructure. The dataset is comprised of 53 patients (15 for training, 38 for testing). The data were acquired on 1.5T or 3T scanners, in 4 medical centers. For each patient examination, the dataset contains 7 manual segmentations and 5 magnetic resonance (MR) sequences as follows: 3D FLAIR, 3D T1-w, 3D T1-w GADO, 2D DP, and 2D T2-w. Both raw and preprocessed images are available. The available preprocessed images were subjected to the following steps: denoising with a non-local means algorithm [10], rigid registration of each image to FLAIR image [11], brain extraction via volBrain platform [12], and bias field correction through the N4 algorithm [13].

Rather than using the available preprocessed images, we employed an alternative preprocessing scheme. Our preprocessing routine yielded better values for the objective function in a preliminary evaluation. For the competition, we preprocessed the raw images according to the following steps: non-uniformity correction with N3 [14]; using mni_autoreg tool [15], intra-subject registration to FLAIR image and inter-subject registration to the MNI152 linear model [16]; brain extraction via bet2 tool [17]. The images were smoothed with a Gaussian filter. To make intensities comparable among acquisitions, the foreground intensities were transformed to have median 0 and interquartile range 1.

Evaluation Criteria In order to compute evaluation metrics, the gold standard is defined as the consensus among 7 manual segmentations through the LOP-STAPLE algorithm [18]. Algorithm assessment will occur in three categories: segmentation, lesion detection, and computational performance. The segmentation assessment involves the Dice similarity coefficient and the average surface distance. The evaluation for lesion detection considers region count metrics as the lesion-wise F1-score. Finally, the computational evaluation includes runtime and memory load as metrics.

3 Evaluation-Oriented Training through Surrogate Metrics

3.1 Segmentation Model and Optimization Method

We adopt a simple multilayer perceptron (MLP) model with a single hidden layer containing just a few neurons for fast computation of outputs. We employ particle swarm optimization (PSO) for training MLP as an alternative to backpropagation algorithm [19]. The backpropagation restricts the form of learning target. On the other hand, PSO enables to use different learning targets without being necessary to develop a specific procedure for each function to be optimized.

3.2 Computationally Cheap Learning Targets

The 3D computation of learning targets would be impractical for iterative training. To reduce the computational burden, instead of costly 3D metrics, we employed cheaper 2D surrogates. The 2D surrogates are calculated in lower dimension and averaged over a reduced number of selected slices. Thus, surrogate metrics are faster due to problem reduction, regardless of hardware-based acceleration. For the MS segmentation challenge 2016, we use a combination of three metrics as learning targets: ρ_1: *Dice similarity score*; two lesion count metrics, ρ_2: *lesion-wise true positive rate* and ρ_3: *lesion-wise positive predictive value*. On model training, we do not target surface-based metrics because algorithms will be ranked just according to ρ_1, ρ_2, and ρ_3 in the competition. Thus, the learning target for evaluation-oriented training is the 2D-Score $= (\rho_1 + \rho_2 + \rho_3)/3$. Note that the three component metrics are in the range [0, 1].

3.3 Oriented Objective Function

In evaluation-oriented training, the objective function (Algorithm 1) has as learning target the 2D-score. Moreover, the oriented objective computes an average of three 2D metrics that requires samples of slices for each patient. The objective function includes a pre-classification rule. This rule prevents unnecessary computation of MLP outputs and takes advantage of the fact that lesions appear hyper-intense in FLAIR images (Algorithm 1 at line 5). Notice that such rule requires standardized intensities. Also, note that 2D-score must be maximized, and the PSO must be configured accordingly.

Algorithm 1 Oriented Objective

```
function OBJECTIVE(sliceSamples)
    for each patient do
        for each slice in sliceSamples do
            for each voxel do
                if FLAIR intensity < 0 then
                    assign non-lesion to voxel
                else
                    classify voxel via MLP
                end if
            end for
            compute 2D metrics
        end for
    end for
    compute average 2D metrics
    objective = 2D-score                          ▷ Section (3.2)
end function
```

3.4 Slice Selection

The plane, quantity, and position of the selected slices can influence the final 2D-score. We adopted a simple strategy and selected the approximate medial slices 85, 90, and 95 in the axial plane of the MNI152 space.

3.5 Model Training and Usage

The PSO-MLP training was repeated 10 times using a repeated holdout method for model selection. As input, we use FLAIR, T1, and T2 sequences that were preprocessed according to our routine. To apply a trained model to new images, we selected the MLP that yielded the best objective function value for all entries in the training set. We developed a segmentation pipeline comprised by preprocessing, segmentation (with the selected MLP) and postprocessing. In the postprocessing stage, we employed a lesion probability atlas to remove false-positives occurring in low probability regions. This lesion atlas was built from the MS challenge 2008 data set.

4 Conclusion

We described evaluation-oriented training with surrogate learning targets to train an MLP inside an MS segmentation pipeline. We report the preprocessing steps employed. The pipeline was applied to the images of the MS Segmentation Challenge 2016. In comparison to previous MS segmentation competitions, the 2016 challenge provides a larger and more diverse dataset, besides a single platform to assess the computational performance of different segmentation methods.

References

1. Styner, M., Lee, J., Chin, B., Chin, M.S., huong Tran, H., Jewells, V., Warfield, S.: 3D segmentation in the clinic: A grand challenge II: MS lesion segmentation. In: MICCAI 2008 Workshop. pp. 1–5 (2008)
2. Barkhof, F., Filippi, M., Miller, D.H., Scheltens, P., Campi, A., Polman, C.H., Comi, G., Ader, H.J., Losseff, N., Valk, J.: Comparison of MRI criteria at first presentation to predict conversion to clinically definite multiple sclerosis. Brain 120(11), 2059–2069 (1997)
3. Souplet, J.C., Lebrun, C., Ayache, N., Malandain, G., et al.: An automatic segmentation of T2-FLAIR multiple sclerosis lesions. In: The MIDAS Journal-MS Lesion Segmentation (MICCAI 2008 Workshop) (2008)
4. García-Lorenzo, D., Prima, S., Arnold, D.L., Collins, D.L., Barillot, C.: Trimmed-likelihood estimation for focal lesions and tissue segmentation in multisequence MRI for multiple sclerosis. Medical Imaging, IEEE Transactions on 30(8), 1455–1467 (2011)
5. Tomas-Fernandez, X., Warfield, S.: A model of population and subject (MOPS) intensities with application to multiple sclerosis lesion segmentation. Medical Imaging, IEEE Transactions on 34(6), 1349–1361 (June 2015)
6. Jesson, A., Arbel, T.: Hierarchical MRF and random forest segmentation of MS lesions and healthy tissues in brain MRI. In: The Longitudinal MS Lesion Segmentation Challenge (2015)
7. Taha, A.A., Hanbury, A.: Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool. BMC medical imaging 15(1), 29 (2015)
8. He, L., Chao, Y., Suzuki, K.: Two efficient label-equivalence-based connected-component labeling algorithms for 3-D binary images. Image Processing, IEEE Transactions on 20(8), 2122–2134 (2011)
9. Santos, M.M., Diniz, P.R., Silva-Filho, A.G., Santos, W.P.: Evaluation-oriented training via surrogate metrics for multiple sclerosis segmentation. In: Medical Image Computing and Computer-Assisted Intervention - MICCAI 2016 (to appear)
10. Coupé, P., Yger, P., Prima, S., Hellier, P., Kervrann, C., Barillot, C.: An optimized blockwise nonlocal means denoising filter for 3-d magnetic resonance images. IEEE transactions on medical imaging 27(4), 425–441 (2008)
11. Commowick, O., Wiest-Daesslé, N., Prima, S.: Block-matching strategies for rigid registration of multimodal medical images. In: 2012 9th IEEE International Symposium on Biomedical Imaging (ISBI). pp. 700–703. IEEE (2012)
12. Manjon, J.V., Coupé, P.: volbrain: An online mri brain volumetry system. In: Organization for Human Brain Mapping'15 (2015)
13. Tustison, N.J., Avants, B.B., Cook, P.A., Zheng, Y., Egan, A., Yushkevich, P.A., Gee, J.C.: N4itk: improved n3 bias correction. IEEE transactions on medical imaging 29(6), 1310–1320 (2010)
14. Sled, J.G., Zijdenbos, A.P., Evans, A.C.: A nonparametric method for automatic correction of intensity nonuniformity in mri data. IEEE transactions on medical imaging 17(1), 87–97 (1998)
15. Collins, D.L., Neelin, P., Peters, T.M., Evans, A.C.: Automatic 3d intersubject registration of mr volumetric data in standardized talairach space. Journal of computer assisted tomography 18(2), 192–205 (1994)
16. Mazziotta, J., Toga, A., Evans, A., Fox, P., Lancaster, J., Zilles, K., Woods, R., Paus, T., Simpson, G., Pike, B., et al.: A probabilistic atlas and reference system for the human brain: International consortium for brain mapping (icbm). Philosophical

Transactions of the Royal Society of London B: Biological Sciences 356(1412), 1293–1322 (2001)
17. Jenkinson, M., Pechaud, M., Smith, S.: Bet2: Mr-based estimation of brain, skull and scalp surfaces. In: Eleventh annual meeting of the organization for human brain mapping. vol. 17, p. 167. Toronto, ON (2005)
18. Akhondi-Asl, A., Hoyte, L., Lockhart, M.E., Warfield, S.K.: A logarithmic opinion pool based staple algorithm for the fusion of segmentations with associated reliability weights. IEEE transactions on medical imaging 33(10), 1997–2009 (2014)
19. Eberhart, R.C., Shi, Y.: Computational Intelligence - Concepts to Implementations. Elsevier (2007)

APÊNDICE C – ARTIGO NA REVISTA *SCIENTIFIC REPORTS*

Reprodução do Registro Bibliográfico para Identificação

COMMOWICK, O.; ISTACE, A.; KAIN, M.; LAURENT, B.; LERAY, F.; SIMON, M.; POP, S. C.; GIRARD, P.; AMÉLI, R.; FERRÉ, J.-C.; KERBRAT, A.; TOURDIAS, T.; CERVENANSKY, F.; GLATARD, T.; BEAUMONT, J.; DOYLE, S.; FORBES, F.; KNIGHT, J.; KHADEMI, A.; MAHBOD, A.; WANG, C.; MCKINLEY, R.; WAGNER, F.; MUSCHELLI, J.; SWEENEY, E.; ROURA, E.; LLADÓ, X.; SANTOS, M. M.; SANTOS, W. P.; SILVA-FILHO, A. G.; TOMAS-FERNANDEZ, X.; URIEN, H.; BLOCH, I.; VALVERDE, S.; CABEZAS, M.; VERA-OLMOS, F. J.; MALPICA, N.; GUTTMANN, C.; VUKUSIC, S.; EDAN, G.; DOJAT, M.; STYNER, M.; WARFIELD, S. K.; COTTON, F.; BARILLOT, C. Objective evaluation of multiple sclerosis lesion segmentation using a data management and processing infrastructure. *Scientific Reports*, v. 8, n. 1, p. 13650, 2018. ISSN 2045-2322.

SCIENTIFIC REPORTS

OPEN

Objective Evaluation of Multiple Sclerosis Lesion Segmentation using a Data Management and Processing Infrastructure

Received: 20 April 2018
Accepted: 6 August 2018
Published online: 12 September 2018

Olivier Commowick[1], Audrey Istace[2], Michaël Kain[1], Baptiste Laurent[3], Florent Leray[1], Mathieu Simon[1], Sorina Camarasu Pop[4], Pascal Girard[4], Roxana Améli[2], Jean-Christophe Ferré[5,1], Anne Kerbrat[6,1], Thomas Tourdias[7], Frédéric Cervenansky[4], Tristan Glatard[8], Jérémy Beaumont[1], Senan Doyle[9], Florence Forbes[9,10], Jesse Knight[11], April Khademi[12], Amirreza Mahbod[13], Chunliang Wang[13], Richard McKinley[14], Franca Wagner[14], John Muschelli[15], Elizabeth Sweeney[15], Eloy Roura[16], Xavier Lladó[16], Michel M. Santos[17], Wellington P. Santos[18], Abel G. Silva-Filho[17], Xavier Tomas-Fernandez[19], Hélène Urien[20], Isabelle Bloch[20], Sergi Valverde[16], Mariano Cabezas[16], Francisco Javier Vera-Olmos[21], Norberto Malpica[21], Charles Guttmann[22], Sandra Vukusic[2], Gilles Edan[6,1], Michel Dojat[23], Martin Styner[24], Simon K. Warfield[19], François Cotton[2] & Christian Barillot[1]

We present a study of multiple sclerosis segmentation algorithms conducted at the international MICCAI 2016 challenge. This challenge was operated using a new open-science computing infrastructure. This allowed for the automatic and independent evaluation of a large range of algorithms in a fair and completely automatic manner. This computing infrastructure was used to evaluate thirteen methods of MS lesions segmentation, exploring a broad range of state-of-theart algorithms, against a high-quality database of 53 MS cases coming from four centers following a common definition of the acquisition protocol. Each case was annotated manually by an unprecedented number of seven different experts. Results of the challenge highlighted that automatic algorithms, including the recent machine learning methods (random forests, deep learning, ...), are still trailing human expertise on both detection and delineation criteria. In addition, we demonstrate that computing a statistically robust consensus of the algorithms performs closer to human expertise on one score (segmentation) although still trailing on detection scores.

[1]VISAGES: INSERM U1228 - CNRS UMR6074 - Inria, University of Rennes I, Rennes, France. [2]Department of Radiology, Lyon Sud Hospital, Hospices Civils de Lyon, Lyon, France. [3]LaTIM, INSERM, UMR 1101, University of Brest, IBSAM, Brest, France. [4]Univ Lyon, INSA-Lyon, Université Claude Bernard Lyon 1, UJM-Saint Etienne, CNRS, Inserm, CREATIS UMR 5220, U1206, F-69621, Lyon, France. [5]CHU Rennes, Department of Neuroradiology, F-35033, Rennes, France. [6]CHU Rennes, Department of Neurology, F-35033, Rennes, France. [7]CHU de Bordeaux, Service de Neuro-Imagerie, Bordeaux, France. [8]Department of Computer Science and Software Engineering, Concordia University, Montreal, Canada. [9]Pixyl Medical, Grenoble, France. [10]Inria Grenoble Rhône-Alpes, Grenoble, France. [11]Image Analysis in Medicine Lab, School of Engineering, University of Guelph, Guelph, Canada. [12]Image Analysis in Medicine Lab (IAMLAB), Ryerson University, Toronto, Canada. [13]School of Technology and Health, KTH Royal Institute of Technology, Stockholm, Sweden. [14]Department of Diagnostic and Interventional Neuroradiology, Inselspital, University of Bern, Bern, Switzerland. [15]Johns Hopkins Bloomberg School of Public Health, Baltimore, MD, USA. [16]Research institute of Computer Vision and Robotics (VICOROB), University of Girona, Girona, Spain. [17]Centro de Informática, Universidade Federal de Pernambuco, Pernambuco, Brazil. [18]Depto. de Eng. Biomédica, Universidade Federal de Pernambuco, Pernambuco, Brazil. [19]Computational Radiology Laboratory, Department of Radiology, Children's Hospital, 300 Longwood Avenue, Boston, MA, USA. [20]LTCI, Télécom ParisTech, Université Paris-Saclay, Paris, France. [21]Medical Image Analysis Lab, Universidad Rey Juan Carlos, Madrid, Spain. [22]Center for Neurological Imaging, Department of Radiology, Brigham and Women's Hospital, Boston, MA, USA. [23]Inserm U1216, University Grenoble Alpes, CHU Grenoble, GIN, Grenoble, France. [24]Department of Computer Science, University of North Carolina, Chapel Hill, NC, USA. Correspondence and requests for materials should be addressed to O.C. (email: Olivier.Commowick@inria.fr)

Multiple Sclerosis (MS) is a chronic inflammatory disease of the central nervous system affecting around 2.5 million persons worldwide, with a prevalence rate of 83 per 100000 (higher rates in countries of the northern hemisphere) and a woman:man ratio of around 2.0[1]. It is characterized by widespread inflammation, focal demyelination, and a variable degree of axonal loss. With the appearance of new treatment molecules modifying the disease evolution (disease modifying drugs - DMD), one of the major challenges in treating multiple sclerosis is now to overcome classical clinical criteria, such as the expanded disability status scale (EDSS), to go towards more sensitive and specific criteria. In this context, Magnetic Resonance Imaging (MRI) plays an important role for the diagnosis[2] and evaluation of the evolution of the disease, thus providing insights to adapt the treatment to each individual due to the highly variable nature of the MS disease course[3].

In this context, the number and spread of lesions in the patient's parenchyma (and their evolution)[2] has become a crucial information on the patient's disease status, which may then be used for validating the patient treatment. This task however requires the delineation of MS lesions: a tedious, manual operation performed by the radiologist. In addition, this delineation is prone to inter-expert variability, especially when the images being used for segmentation differ from a center to another (in terms of protocols, modalities and intrinsic MRI quality). Doing this task manually on large databases of patients is therefore almost impossible and automatic algorithms, thoroughly validated, have become a crucial need for the clinical community. To simplify the clinician's task, a large literature of automatic segmentation methods has been devised[4–6] with a large spectrum of algorithms from classical tissue intensity classification and lesion modeling to machine learning.

All published approaches are however evaluated on different datasets, usually not calibrated, and their results are therefore usually not directly comparable, making difficult the choice of the most relevant method adapted to a clinical context. To overcome this issue, competitions (so-called challenges) have been organized in MS lesion segmentation in the past years. The first one was organized at the MICCAI 2008 conference[7]. It evaluated nine different methods on a database of 45 patient images (from two different centers: 20 for training and 25 for testing), with respect to a ground truth composed of two expert segmentations for each case. However, no protocol standardization was performed between the two sites, therefore two raters were not enough to handle the variability in the acquired images and get a sufficiently reliable consensus manual segmentation. The second major challenge on MS lesion segmentation was held in 2015 at the IEEE ISBI international conference[8]. It was more focused on the study of longitudinal lesion evolution with specific evaluation metrics based on segmentation volume evolution (in addition to the regular segmentation overlap metrics used in 2008). This challenge evaluated 10 different methods on a dataset composed of five patients images each with an average of 4.4 time points, each time point being manually delineated by two experts. As for the MICCAI 2008 challenge, two raters were not enough to account for disparities in the different raters manual segmentations and get a representative consensus. The process of evaluation was similar for the two challenges. A subset of the patient images was provided to the participants with the ground truth (GT) segmentation to the participants for them to train their respective methods. In a second step, a testing set was provided (without the ground truth) to the participants asking them to submit back their results. Evaluation was then performed on those results using overlap-based metrics.

Several problems may however affect such challenges. First, as a general comment for all challenges, a lack of fairness may exist between the participants: since the testing images are provided, some participants may indeed optimize the parameters of their algorithms on a patient basis to obtain better results. Doing this illustrates the potential of the method but not its practical usability: a clinician would prefer to use always the same set of parameters to process each new or returning patient. In addition, since participants run their algorithm on their own computing environment, no evaluation relative to computing performance (e.g. required memory of computing time) is possible. There is therefore a need for computing platforms for supporting challenges including data storage, processing pipelines (i.e. segmentation algorithms work-flow used) integration and evaluation on stored datasets. Such platforms would provide a truly fair comparison between fully automatic methods. In addition, such remote computing platforms, able to host a large variety of algorithms, announce what the future cloud computing services will provide to assist clinicians (radiologists, neurologists, …) in using computer aided diagnosis solutions. This computing environment also opens the road to open-science platforms where people will find solutions to post their data, send or retrieve algorithmic solutions and provide an independent yet secure environment to compare, assess and combine various algorithms outcomes and solve clinical problems.

Another issue in segmentation challenges is the number of manual delineations to compute the ground truth. Usually only two are available, which is insufficient to illustrate the inter-expert variability, particularly when considering MS lesions segmentation. Finally, and specifically to MS lesions segmentation, previous challenges considered only segmentation based metrics, ignoring the number of correctly detected lesions independently of their shape, which is an acute criterion to assess the disease evolution[2]. This would be very beneficial for the clinician, especially when considering MS evolution where the number of new lesions is critical.

We proposed and organized in 2016 a new generation of segmentation challenge hosted at the MICCAI international conference (http://www.miccai2016.org). It aimed at proposing solutions to several of the previously mentioned defects first by gathering an unprecedented database of MS patients, coming from three different centers (representing four different scanners, one of which was intentionally hidden at the training phase from the challengers to test their algorithms' adaptation capabilities) but all following a common consensus protocol[9], each patient being delineated by seven experts to evaluate not only automatic methods performance but also inter-expert variability of manual segmentation. We have performed the evaluation on a dedicated computing platform provided by France Life Imaging (https://www.francelifeimaging.fr/en), providing pipeline integration, database storage and automatic execution capabilities. Challenge participants were asked to train their algorithms on a reduced set ($n = 15$) and then integrate their pipeline on the platform, requiring no action from them in the latter parts of the evaluation process ($n = 38$). We also proposed an evaluation strategy on two separate levels: a segmentation level where the overlap precision of the segmentation was evaluated; and a detection level where the number of correctly detected lesions was evaluated, independently of the precision of their shape.

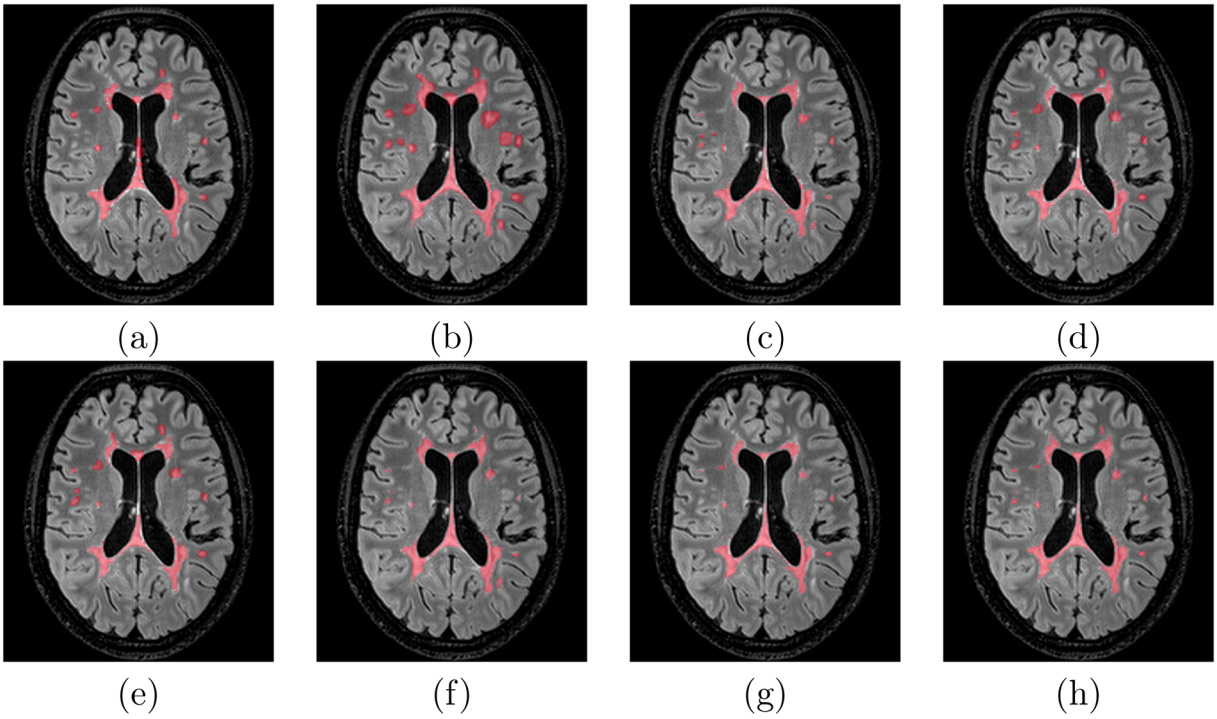

Figure 1. Illustration of an MS patient delineations overlaid on the 3D FLAIR image. (**a**–**g**) Individual manual delineations of MS lesions from each of the experts, (**h**) consensus segmentation considered as the ground truth.

We present in this article a retrospective analysis of this challenge and the methods we used to obtain those results. The main outcomes of the challenge highlighted that automatic algorithms are still trailing human expertise on the front of MS lesions segmentation and sensitive to unknown images (different scanners) even with an harmonized acquisition protocol. This happens for all methods, independently of their category (recent machine learning algorithms including deep learning or random forests or more classical tissue classification algorithms). In addition, we demonstrate how using an open-science computing environment allows for the combination of multiple algorithmic outcomes, and how combining these algorithms could lead to improvements in detection and contouring of MS lesions. Together with the computing platform introduced in this paper, this could lead to tremendous help for the clinicians in the use of automatic segmentation algorithms to support their diagnosis and treatment follow-up in MS.

Results

Challenge data, computing platform and participating teams. The first major result of this study is the gathering of a database of 53 multiple sclerosis patients with "ground truth" of very high-quality. The database patient scans were following the OFSEP protocol recommendations in[9], which is currently applied in France for the constitution of the national cohort in MS (for more details on the protocol, see Section 4.1). Following this approach has allowed for an evaluation representative of the current imaging protocols standards and easily usable to characterize the best performing algorithm for future use. This standardization of imaging protocols announces how the dissemination of computer aided diagnosis and imaging biomarkers solutions will be implemented in the future. Image processing algorithms indeed need image normalization and quality control to ensure peak performance. In addition, the images came from three different sites in France on four different MRI scanners and different manufacturers (Siemens, Philips and GE) including three 3 T and one 1.5 T magnets. For each MS patient case, an unprecedented number of seven manual delineations was gathered, from trained experts split over the three sites providing MR images. From these segmentations, a consensus "ground truth" segmentation was built for evaluation with the LOP STAPLE algorithm[10]. We present in Fig. 1 an example of a patient 3D FLAIR, the seven manual segmentations of lesions and their consensus segmentation, illustrating the variability for a representative patient between expert segmentations. Patients demographic data were the following: average age of 45.3 years ($\pm$10.3 years) with a male:female ratio of 0.4. This database was then split into two sets: one training set of 15 patients from three scanners (thus intentionally missing one scanner from the database) given to participants, and one testing set of 38 patients, not seen by the participants, used for evaluation. Demographics of patients do not vary significantly over the different sitesin terms of age. Some variations exist in the male:female ratios in some centers. The training and testing sets have an average age difference of 5 years (training set patients are 5 years younger).

A total of thirteen teams were evaluated, and the website (http://portal.fli-iam.irisa.fr/msseg-challenge/), databases and algorithms will remain open for future use. A summary of the evaluated methods is presented in Table 1 with a short description of their characteristics (MR sequences used as input, implementation, main

Team	Authors	Segmentation approach	Platform	Sequences used
1	J. Beaumont O. Commowick	Graph cut segmentation initialized by a robust EM[23,24]	CPU	T_1-w, T_2-w, FLAIR (preprocessed)
2	J. Beaumont O. Commowick	Multi-modal abnormalities detection from normalized images on an atlas[25,26]	CPU	T_2-w, FLAIR (preprocessed)
3	S. Doyle F. Forbes	HMRF segmentation framework with a weighted data model[27,28]	CPU	T_1-w, FLAIR (raw)
4	J. Knight A. Khademi	Segmentation by edge-based model of partial volume/pure tissue gray levels[29,30]	CPU	FLAIR (raw)
5	A. Mahbod C. Wang	Supervised artificial neural network with intensity and spatial based features[31,32]	CPU	FLAIR (preprocessed)
6	R. McKinley T. Gundersen	Ensemble of three 2D fully Convolutional Neural Networks with skip connections[33]	GPU	FLAIR (preprocessed)
7	J. Muschelli E. Sweeney	Random Forest (RF) on normalized multi-modal features[34]	CPU	T_1-w, T_2-w, PD, FLAIR (raw)
8	E. Roura X. Lladó	Outlier segmentation based on brain tissue labeling and post-processing rules[35,36]	CPU	T_1-w, FLAIR (raw)
9	M. Santos A. Silva-Filho	Multilayer perceptron with cost functions oriented to competition evaluation metrics[37,38]	CPU	T_1-w, T_2-w, FLAIR (preprocessed)
10	X. Tomas-Fernandez S.K. Warfield	Lesions and brain tissue segmentation through simultaneous estimation of spatially and population varying intensity distributions[39,40]	CPU	T_1-w, T_2-w, FLAIR (raw)
11	H. Urien I. Bloch	Hierarchical segmentation using max-tree, spatial context and anatomical constraints[41,42]	CPU	T_1-w, T_1-w Gd, T_2-w, PD, FLAIR (raw, preprocessed)
12	S. Valverde M. Cabezas	Cascade of two 7-layer convolutional neural networks of 3D patches[43]	GPU	T_1-w, T_2-w, PD, FLAIR (preprocessed)
13	F.J. Vera-Olmos N. Malpica	Grey matter filter as input to a RF classifier corrected with Markov Random Field processing[44]	CPU	*T*1-w, *T*2-w, PD, FLAIR (preprocessed)

Table 1. MS lesion segmentation methods evaluated at the MICCAI 2016 challenge.

methodology). The algorithms evaluated in the challenge are representative of a broad range of the available methods in the recent literature, with unsupervised tissue classification methods, level-sets, random forests and deep learning (convolutional neural network, artificial neural networks). Depending on the challenger team, the image modalities used for the segmentation varied from just one (usually FLAIR) to all provided modalities. Most evaluated algorithms ran on regular computer CPU, while two (team 6 and 12) leveraged specific hardware (GPUs) for intensive computation (e.g. deep learning). The computing infrastructure was able to provide the relevant computing solution for all requirements.

These teams were evaluated in a distributed Web platform based on software containers provided by the France Life Imaging platform, allowing for the automatic, challenger independent evaluation of the algorithms (see Section 4.3 for more details on the challenge execution platform). Using such a platform, providing integrated storage and computing facilities for challenges, allowed for fair comparisons as challengers could not tune their algorithm specifically for each test patient. Each challenger was indeed asked only to provide a binary image (i.e. an annotated Docker container image) of their processing pipeline and the evaluation was later on run automatically on the platform with the following metrics used.

Two kinds of performance metrics were set up for evaluation. Clinicians evaluate lesion segmentation in multiple sclerosis with different criteria. Lesion segmentation precision, i.e. the precision of contours delineated for each lesion, is crucial as the total volume of lesions (total lesion load - TLL) is part of the criteria to evaluate disease severity[11,12]. When coming to pathology evolution or treatment efficiency evaluation however, lesion count and particularly the number of new lesions independently of their sizes is key. Moreover, this lesion count is a crucial component of MS diagnosis according to McDonald criteria[2]. For these tasks, detecting all lesions is more important than their precise contours. We have therefore implemented a large set of evaluation measures for the challenge with the goal of evaluating these different aspects. Evaluation in the following is therefore split into three major categories of evaluation metrics:

- Segmentation evaluation: does the algorithm provide a precise delineation of each lesion? This category includes average surface distance and Dice overlaps as the main metrics
- Lesion detection evaluation: does the algorithm find all lesions in the image independently of its precise delineation? This category includes the F_1 score, gathering in one scalar information on the number of lesions correctly and incorrectly detected.

All methods are outperformed by the experts. We have automatically clustered the average algorithms and experts annotations agreements (with their covariances accounted for) with respect to the "ground truth" (see Section 4.4 for more details). Results of this clustering, illustrated in Fig. 2 for all couples of measures considered in the challenge, highlight a major result of the challenge: over all patients and all evaluation metrics, each individual method performs slightly below all experts. On all graphs in Fig. 2, all experts (and only them) are indeed

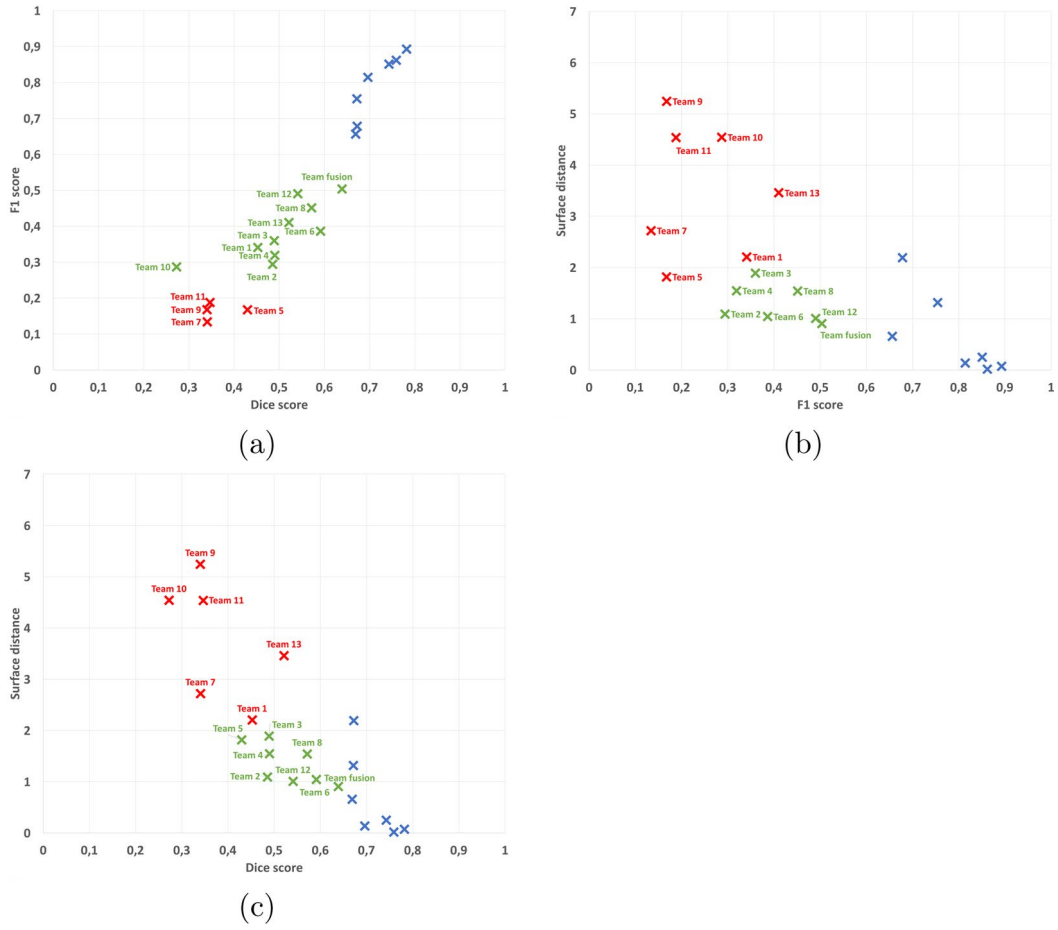

Figure 2. Graphical results illustration of automatic clustering of average results for each team and expert into three groups (scatter plots of pairs of two evaluation parameters: (**a**) Dice and F_1 scores, (**b**) surface distance and F_1 scores, (**c**) surface distance and Dice scores). Legend: blue crosses: group 1 (always containing only the seven experts even though the clustering is automatic), green crosses: group 2 (best performing algorithms), red crosses: group 3 (lower "quality" algorithms). Team numbers associated with each point on the graph are indicated as labels. Team fusion indicates a composite segmentation result further discussed in Section 2.5.

always grouped in a single cluster that performs better than all automatic algorithms. Two other clusters are also distinguished in these graphs, which vary depending on the evaluation metric, that regroup better performing and lower performing algorithms for each couple of evaluation metrics.

In those graphs, we can additionally study the performance of automatic algorithms with regards to each evaluation metric considered (average surface distance, Dice score and F_1 score). Automatic methods fail much more on the detection of lesions (F_1 score), with a minimum average score of 0.13 and maximum average of 0.49, while the minimum average score obtained by an expert is 0.66 (significant difference, Wilcoxon signed rank test, $p = 3.7 \times 10^{-5}$). This is understandable however as all algorithms are primarily designed to obtain the best segmentation scores while not considering lesion detection which is a somewhat different task. However, even on the Dice score, which is a segmentation metric, the best automatic method performs lower than the lowest expert average score: it reaches an average of 0.59 while the lowest expert is on average at 0.67 (significant difference, Wilcoxon signed rank test, $p = 2.9 \times 10^{-3}$). The average surface distance is a more balanced metric in terms of results with the second group of algorithms in each graph reaching the level of agreement that the experts do with the consensus.

Segmentation on an unknown scanner leads to poorer performance. Scanner 3 in the testing database was unknown to the teams participating to the challenge. On this center, we have evaluated how automatic algorithms performed without knowing the image characteristics beforehand. The results of this comparison highlight for a large number of automatic algorithms a slight decrease in performance when encountering unknown images, even if they come from a common protocol. This evaluation per center and per evaluation metric is presented in Fig. 3.

Looking closer at the graphs in Fig. 3, we can observe for the detection metric (F_1 score, Fig. 3b) a slight decrease for 8 teams among the thirteen evaluated, leading to an average score over all teams of 0.22 for center 3 while the same automatic methods range between 0.32 and 0.39 for other centers. The same trend can be observed

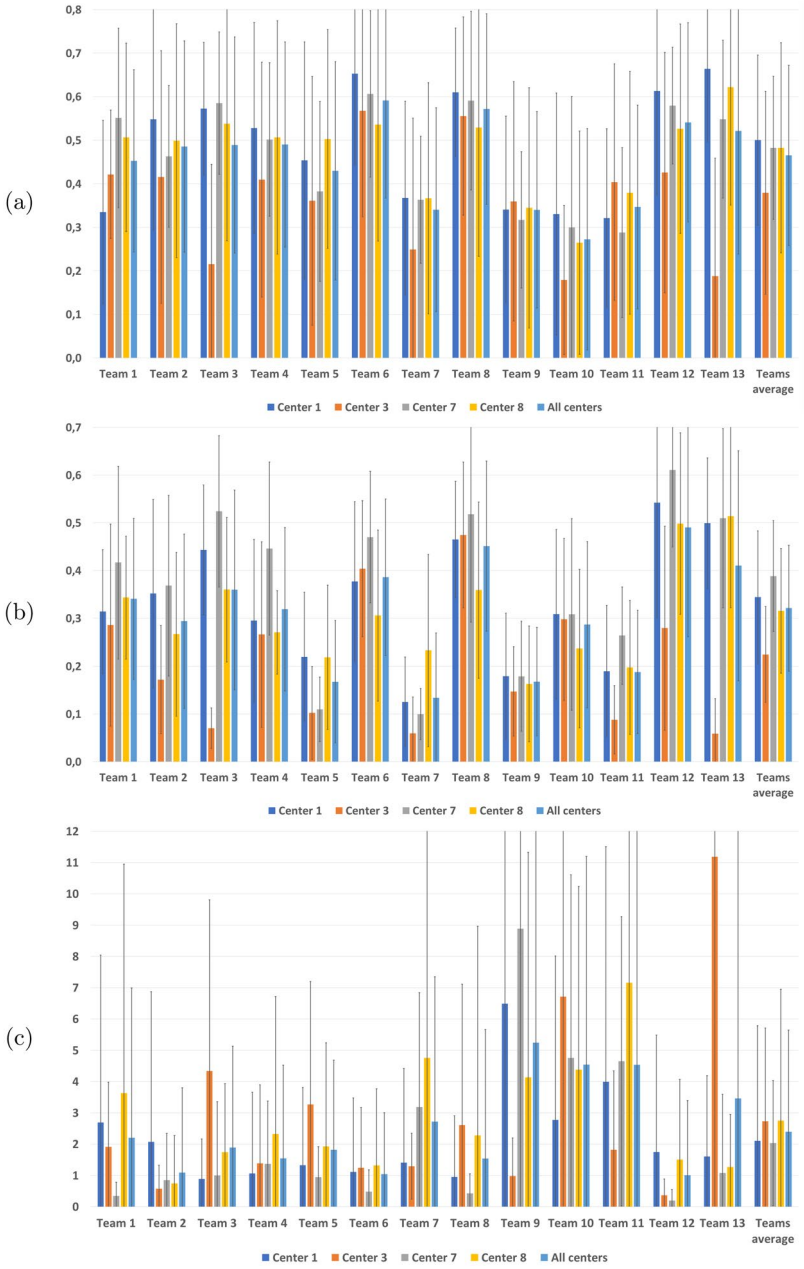

Figure 3. Dice scores (**a**) F_1 scores (**b**) and average surface distances (**c**) with respect to the consensus per team for each center and averaged over all centers.

for the Dice score segmentation metric (Fig. 3a) with a slight decrease in performance for 8 teams as well, and an average score over all teams of 0.38 while the same algorithms reach a level ranging between 0.48 and 0.50 on other centers. The observations are different for the average surface distance (Fig. 3c), where results for center 3 are at the same level as center 8, however the variance in results is much higher, preventing from finding any statistically significant difference on that metric.

Combining methods through label fusion improves over individual algorithms. In addition to the individual automatic algorithms, we have evaluated a composite team named "team fusion". This method gathered the other thirteen teams segmentations in a consensus through label fusion using the LOP STAPLE algorithm[10]. The goal of this fourteenth method was to evaluate the capability of such a label fusion method to overpass the individual difficulties of each method and thus obtain results closer to the ground truth. We present the results of this evaluation on the different evaluation metrics in Fig. 4.

This composite algorithm improves the average results the average results of individual automatic algorithms for all metrics, suggesting its ability to incorporate the best of each team into a consensus segmentation, better in line with the experts. These results are confirmed by points "Team fusion" in the clustering graphs in Fig. 2.

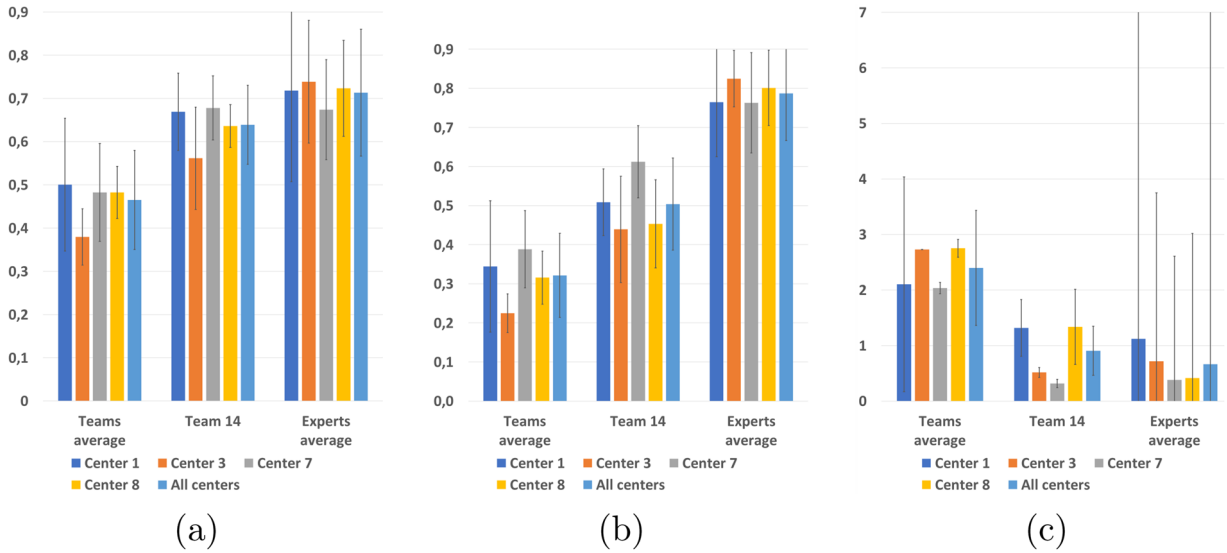

Figure 4. Dice scores (**a**) F_1 scores (**b**) and average surface distances (**c**) with respect to the consensus for each center and averaged over all centers for composite Team fusion with respect to the average experts agreement level.

However, the results obtained are still not perfect and lag behind the experts level of agreement with the "ground truth". More precisely, the improvement of team fusion over other algorithms is particularly visible on segmentation metrics (Dice scores and average surface distance) since it provides segmentation performances similar to the lowest experts. This improvement is however less important on the detection metric (F_1 score). This smaller improvement seems logical as the label fusion algorithm used for team fusion is primarily designed to optimize segmentation performance and not specifically detection. With that said, the first position of Team fusion among the segmentation methods illustrates how a composite algorithm mixing results of other teams is able to perform better than each individual automatic method. This also illustrates the importance to provide an open-science computing platform able to combine results of independent algorithms.

Lesion load and lesion size directly influences automatic segmentation quality. We additionally performed an experiment to evaluate, independently of their individual behaviors, the algorithms sensitivity to the true amount of lesions in the "ground truth" for a given patient. To this end, we averaged the Dice scores (respectively the average surface distances and F_1 scores) over all methods for each patient and plotted in Fig. 5 this average value with respect to either the number of lesions or the total lesion load in the consensus. On each graph, we then computed a log-linear regression for which we display the Spearman squared correlation.

From Fig. 5, it is clear that the worst results are obtained for patients whose total lesion load is low. This is especially true for segmentation performance scores: the Dice score (the squared correlation R^2 of the regression reaches 0.82) and the average surface distance (R^2 of 0.71). For the F_1 score (a detection metric), the correlation is however weaker than for the total lesion load (R^2 of 0.45). From these graphs, the correlation between the number of lesions and the obtained scores is less clear, all correlations being smaller than with the total lesion load (R^2 of 0.46 with the Dice score, 0.38 with the F_1 score, and 0.60 with the average surface distance). This result however seems reasonable since a patient presenting many small lesions is intuitively more difficult to delineate than a patient with a small number of large lesions.

Total lesion load in a patient is thus very correlated with segmentation and detection scores while not with the number of lesions. To further qualify this fact, we performed an experiment considering detection scores individually for each lesion in regard of its volume. We have thus computed, for each team and for each lesion of the "ground truth" of each patient, a binary detection score telling whether the lesion was detected or not by a specific team. Counting the number of teams which detected the lesion thus provides us with a rate of detection for each lesion (a rate of 0% meaning that no team detected the lesion, and 100% meaning that all teams detected the lesion). Those detection rates were further binned according to lesion volume. The graph in Fig. 6 illustrates their relation with respect to lesion volume (in mm^3).

From Fig. 6, we can clearly see that not only total lesion load influences segmentation quality, but lesion volume is also clearly linked with lesion detection (R^2 of 0.88 after a logarithmic linear regression). All methods tend to fail (rates of detection going to zero) for small lesions, while almost all teams are detecting the lesions when their volume is sufficiently large.

Delineating an image with empty consensus. We finally present in Table 2 results obtained by each expert and each evaluated pipeline on a specific case, from Center 7, where no lesion was present in the consensus segmentation. In this specific case, none of the proposed detection and segmentation measures can be used since they all rely on the fact that the consensus is not empty. We thus instead defined two specific metrics: the number

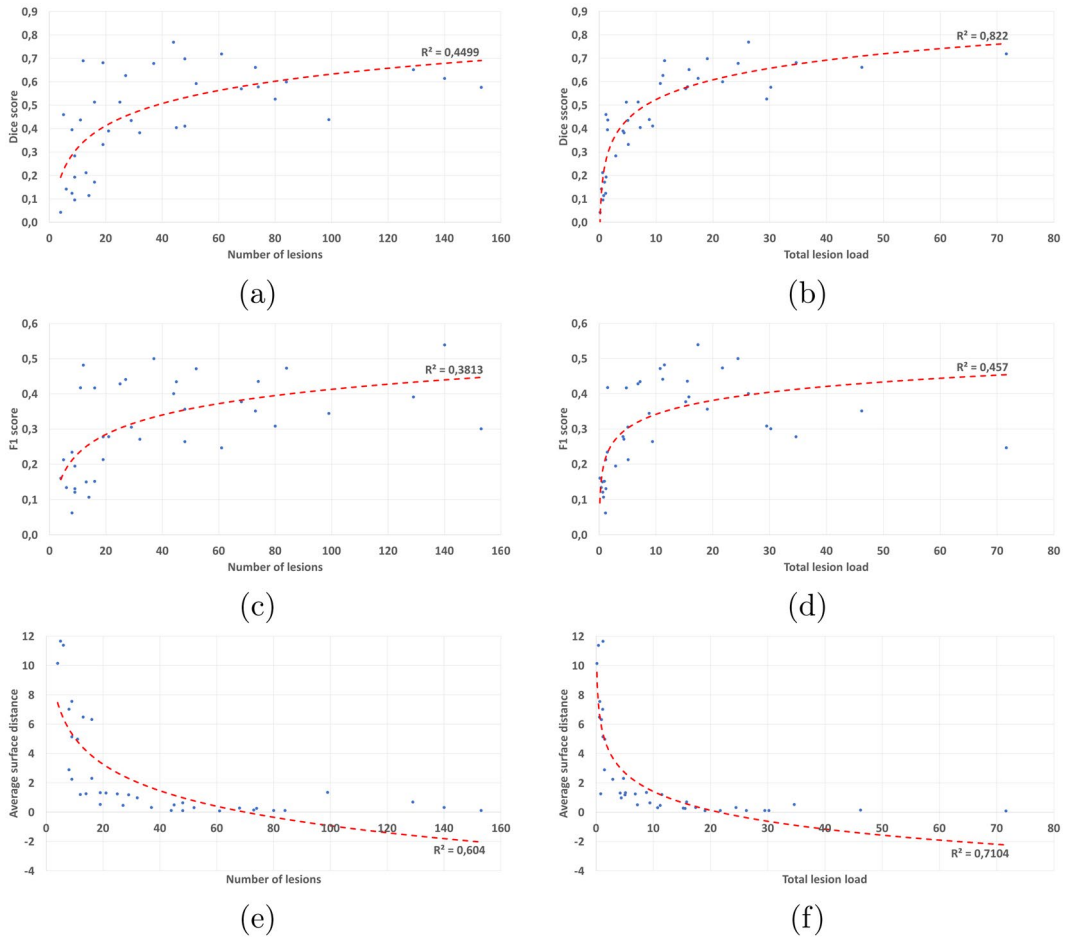

Figure 5. Link between average scores of all methods and number of lesions (first column) and total lesion load (cm^3, second column). First line: Dice score, second line: F_1 score, third line: average surface distance.

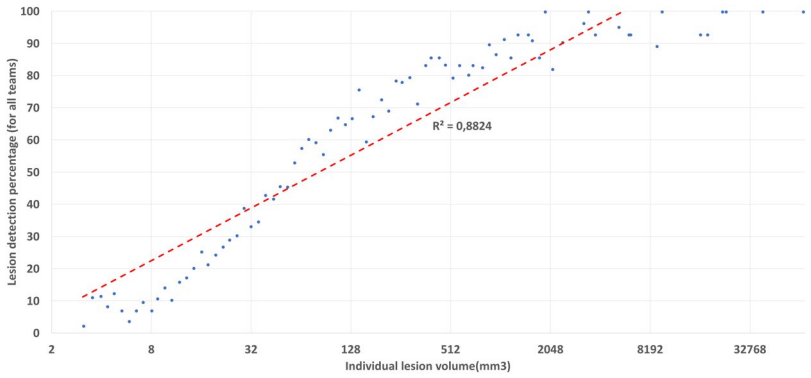

Figure 6. Individual lesion detection rate (average over all methods) as a function of lesion size. X-axis: individual lesion volume on a logarithmic scale. Y-axis: detection rate (in percentages, number of teams detecting a lesion of this volume over all patients).

of lesions detected (i.e. the number of connected components whose size is larger than 3 mm^3 in the segmentation), and the total lesion load (i.e. the total volume of the previously extracted connected components) found by each algorithm. For both metrics, since the consensus contains no lesions, a perfect value is 0 while the results get worse when the metrics grow.

Several observations may be drawn from this table. First of all, among experts, two delineated no lesions while five actually delineated one or several lesions (from one to 8 depending on the expert, and from 0.02 to 11 cm^3). The fact that the consensus is empty therefore means that the experts were not agreeing on the position and extent of lesions, which lead to no lesions in the final "ground truth". Among the automatic segmentation pipelines, the

	Lesions volume (cm^3)	Number of lesions
Expert 1	0.052	2
Expert 2	0.090	2
Expert 3	10.887	8
Expert 4	**0**	**0**
Expert 5	0.017	1
Expert 6	**0**	**0**
Expert 7	0.029	2
Team 1	8.252	18
Team 2	**0**	**0**
Team 3	**0**	**0**
Team 4	NA	NA
Team 5	28.436	522
Team 6	0.473	7
Team 7	5.990	168
Team 8	**0**	**0**
Team 9	2.545	33
Team 10	11.085	31
Team 11	3.436	42
Team 12	0.056	1
Team 13	0.074	4

Table 2. Number of lesions and lesion volume detected by each team and expert on the no consensus lesion case.

results are also largely varying: depending on the team, the number of lesions delineated varies from 0 to 522, while the lesion load detected varies from 0 to 28.44 cm^3. In addition, this image caused problems to some algorithms not initially designed for patients without lesions (team 4). This is an interesting case as it highlights the different behaviors of the algorithms on a case for which the pipelines were not designed. Overall, we can notice that most of the methods behave well in comparison to the experts.

Discussion

We have presented the first challenge based on an integrated computing platform, applied to multiple sclerosis lesions segmentation. The challenge computing platform was constituted of 1- a database to store the challenge images and results from the challengers, 2- a computing platform on which the evaluation was performed independently of the participants who were asked to post their processing pipelines, and 3- an automatic evaluation of the results against the "ground truth". This open-science computing platform has many advantages, including a fair comparison of the participants algorithms being run on the same platform and with the same set of parameters for all patients. In addition, the packaged algorithms may be re-used for other applications or if the validation database gets extended. As future work, we plan at transitioning to use the BIDS format (http://bids.neuroimaging.io) to provide a standardized and more intuitive way of storing both the input data and output results. This would provide a great improvement in easing the pipeline design and integration in the platform.

This platform was put together for the specific organization of a challenge on multiple sclerosis segmentation with a database of 53 patients each with an unprecedented number of seven manual segmentations from trained experts. A total of thirteen teams participated, illustrating the variety of algorithms both in terms of methodology and implementation. All results computed from the challenge were very insightful and revealed several points worth of discussion. First of all, despite that methods vary in their results, the point where a single automatic method is able to perform as well as the consensus of the experts has not yet been reached. The experts are indeed always slightly better than any method for all performance measures. More specifically, all methods perform relatively poorly on detection metrics, which is however an important point for MS diagnostic and clinical evaluation of the patient evolution. Historically all methods have been interested in segmenting well the contours of the lesions rather than counting well the number of lesions. As a consequence the detection metrics are not optimal, which explains why results are well below the experts. On a more positive point, recent methods such as those based on machine learning (especially deep learning) have made great progress and the gap is reducing, which leaves hope to reach the same level of agreement than the experts. In addition, it should be remembered that it is always difficult to define a "ground truth" for MS lesions segmentation. The experts have indeed a relatively large variability, which comes from different appreciations of the image and of the definition of a lesion. Finally on this point, it is also interesting to note that a composite method for segmentation (team fusion) combining the different automatic methods while rejecting outliers is able to drastically improve segmentation results and get closer to the ground truth. However, this happens mainly for segmentation performance metrics and less for detection performance metrics. This may be due to the inherent design of the label fusion methods that do not work on a lesion basis but rather on a voxel basis, and thus favor segmentation based metrics. In addition, since many methods fail at delineating well lesions when the total lesion load is small (see Fig. 5), this composite method is not able to perform a good segmentation for these cases.

We have also illustrated through this challenge the behavior of methods in several specific cases: testing for scanner dependency of the algorithms, where we compared the results for four different scanners, one of them being hidden from the training dataset. This study illustrated that all methods are still sensitive to scanners on which they were not trained for (either training in the machine learning sense or training in the sense of parameters tuning by a human being), obtaining lower scores for those images. On the contrary, when the training set included representative images of different scanners, all algorithms behaved equally independently of the center or scanner. This suggests the importance of looking for more training independent methods or, meanwhile, to have enough representative cases to train on (at the high cost of providing multiple manual image annotations from experts). A second specific case considered a patient with no lesions in the ground truth provided from the experts. For that patient, even if some methods had not planned this case, all methods behaved globally well even though participants were not told in advance about this fact.

While not mentioned explicitly in this article, we also looked at the relation between algorithm performance and preprocessing or modalities used. For both aspects, there is no clear evidence of a link. Some algorithms perform well while using only a subset of modalities, while some other that use all modalities perform less well. However the reverse is also true: some of the best algorithms use all modalities while some less good ones use a smaller number of modalities. This is however a crucial aspect as information on this could help in the design of shorter acquisition protocols in the future, using only those modalities useful for automatic segmentation. Future works, which have to be in close link with the challenge participants (but facilitated thanks to our computing platform), will look at the robustness of results of the individual algorithms with respect to both preprocessing used and modalities used. This will provide a great insight into optimal, fast protocol design and optimal preprocessing.

We have clearly demonstrated in Fig. 5 a link between the total lesion load in a patient and the performance of segmentation methods: the smaller the true total lesion load, the worse the segmentation results were for every metric. As mentioned earlier, this is partly linked to the fact that voxel-based performance metrics are much more sensitive when the number of voxels in the true segmentation is small. However, this is not the only reason: automatic segmentation methods are indeed behaving slightly worse on these cases and a focus on them should probably help in designing algorithms adapted to all situations. This link between performance and lesion load does not generalize to the number of lesions (also seen in the same figure), which illustrates that there is no clear sensitivity of the methods to the number of lesions for a patient. However, this link is clearly related to a correlation between lesion volume and lesion detection rate, as demonstrated in Fig. 6. This further indicates that lesions are clearly less well detected or even not at all when they are small, which seems rather logical as it intuitively seems tougher for an algorithm to properly locate a small lesion than a larger one.

Evaluation metrics presented in this article are a selection per category (segmentation and detection) of the metrics described in Section 4.4 and computed for the challenge. We chose them as being representative and most informative of the main qualities and defaults of the algorithms. Of course, as illustrated for more general segmentation evaluation purpose[13], three metrics may not be enough for describing the behavior of each method in its entirety. As explained in Section 4, we have complemented these three measures with many other complementary measures, for which we encourage the interested reader to look at the supplementary materials (https://doi.org/10.5281/zenodo.1307653).

Methods

We present in the following the methodological details that allowed us to draw the results and conclusions previously outlined. This section is split in several subparts. Sections 4.1 and 4.2 present in more details the evaluation database used in the challenge and the way in which the manual delineations were carried out and averaged into a "ground truth". Section 4.3 then outlines the computing platform used in the challenge which was necessary to guarantee a fair comparison of the algorithms. Finally, Section 4.4 presents the evaluation metrics used in the challenge as well as the analyses plan derived from them in this article.

Reference Images Database. In this segmentation evaluation challenge, we relied on a database of images of 53 multiple sclerosis patients following the OFSEP protocol recommendations in[9], which is currently applied in France for the constitution of the national cohort of MS patients. Following this approach allows for an evaluation representative of the current standards and easily usable for newly acquired images. For our challenge, images came from three different sites in France on four different MRI scanners from different manufacturers (Siemens, Philips and GE) including three 3T and on 1.5T magnets. The repartition of the 53 patients is shown in Table 3. More demographic details on the ages and gender repartitions into the training and testing groups are provided in supplementary material (https://doi.org/10.5281/zenodo.1307653). Overall, no significant difference of age can be seen between the different centers. While gender differences exist between some centers (in particular center 8), we believe this is of little importance with regard to lesion segmentation and detection quality compared to scanner to scanner differences.

These patients were selected to have variable amounts of lesions both in volume and number, and from different centers to represent the variability that may be encountered across sites. For each patient, the following images were provided for each MS patient (see details of the sequence parameters in Table 4): a 3D FLAIR sequence, a 3D T1 weighted sequence pre and post-Gadolinium injection, an axial dual PD-T2 weighted sequence. These patients were then split (see Table 3) between a training and testing datasets. The testing dataset was not made available to the challengers and was used to evaluate the different methods while the training dataset was provided, together with the ground truth segmentations, for challengers to train their algorithms. Images acquired on one center (3) were not part of the training dataset, with the goal of evaluating how much algorithms were dependent on the training set and sensitive to acquisition settings.

Center number	Scanner model and site	Training cases	Testing cases	Age (y.o.)	Gender ratio M:F
1	Siemens Verio 3T (University Hospital of Rennes)	5	10	43.6 ± 12.6	0.36
3	General Electrics Discovery 3T (University Hospital of Bordeaux)	0	8	48.9 ± 11.5	0.14
7	Siemens Aera 1.5T (University Hospital of Lyon)	5	10	45.3 ± 9.8	0.25
8	Philips Ingenia 3T (University Hospital of Lyon)	5	10	45.5 ± 7.8	0.87

Table 3. Demographics data of multiple sclerosis patients collected for the challenge and their repartition among training and testing datasets.

Scanner	Modality	Matrix	Slices	Voxel resolution (mm)
GE Discovery 3T	Sagittal 3D FLAIR Sagittal 3D T1 Axial 2D DP-T2	512 × 512 512 × 512 512 × 512	224 248 From 28 to 44	0.47 × 0.47 × 0.9 0.47 × 0.47 × 0.6 0.43 × 0.43 × 3 Gap: 0.5
Philips Ingenia 3T	Sagittal 3D FLAIR Sagittal 3D T1 Axial 2D PD-T2	336 × 336 336 × 336 512 × 512	261 200 46	0.74 × 0.74 × 0.7 0.74 × 0.74 × 0.85 0.45 × 0.45 × 3
Siemens Aera 1.5T	Sagittal 3D FLAIR Sagittal 3D T1 Axial 2D PD-T2	256 × 224 256 × 256 320 × 320	128 176 25	1.03 × 1.03 × 1.25 1.08 × 1.08 × 0.9 0.72 × 0.72 × 4 Gap: 1.2
Siemens Verio 3T	Sagittal 3D FLAIR Sagittal 3D T1 Axial 2D PD-T2	512 × 512 256 × 256 240 × 320	144 176 44	0.5 × 0.5 × 1.1 1 × 1 × 1 0.69 × 0.69 × 3

Table 4. Acquisition details for each sequence and each scanner for the training and testing MS patients databases.

For each patient, the challenge data includes raw datasets, and preprocessed datasets where the following steps were performed:

- Denoising of each modality using the non local means algorithm[14]
- Rigid registration of each modality on the FLAIR image[15]
- Brain extraction (skull stripping) using the volBrain platform[16], from the T1-w image and applied to other modalities
- Bias field correction of each modality using the N4 algorithm[17].

MS Lesions Ground Truth. Based on manual segmentation on our MS database, we first aimed at getting ground truth segmentations of multiple sclerosis lesions. This task is difficult and variability exists between experts depending on various factors, even when they follow common protocol, depending on many factors (image quality, training, modalities…). We chose to build for this challenge an unprecendented set of seven manual delineations for each patient. These delineations were performed manually on the 3D FLAIR image with control on the T2 weighted image. Each manual segmentation was performed by a trained junior expert, validated and corrected under the supervision of senior radiologists with a long experience in multiple sclerosis. More specifically, a first meeting between senior radiologists and workshop organizers of each site took place to determine the segmentation strategy and adopt a common tool (http://www.itksnap.orgITK-Snap) to perform manual segmentation. Junior radiologists were then recruited on each site and trained by the expert radiologists on a separate training set and when their agreement was above a threshold of 80%, they were allowed to delineate the 53 patient cases. Each case was segmented in isolation of the other cases to limit possible bias. Segmentation experts were split between the three sites which provided the patient images: 4 in Lyon, 2 in Rennes and one in Bordeaux.

MS lesions segmentation is known to be expert- and center-dependent, which can lead to relatively large discrepancies between individual manual segmentations. To cope with this problem, we computed for each patient a consensus segmentation by using the Logarithmic Opinion Pool Based STAPLE (LOP STAPLE) algorithm proposed by[10]. This algorithm computes iteratively, using an Expectation-Maximization approach, a consensus segmentation based on penalties for individual deviations from agreement between manual experts segmentations. This algorithm has several advantages: it is robust to differences between manual expert segmentations, and it allows the computation of agreement scores with respect to the consensus segmentation considered then as ground truth.

Computing Architecture for Automatic MS Lesions Segmentation Evaluation. One of the critical aspects in performing an independent challenge and benchmarking of medical image processing solutions is to provide a unified infrastructure able to:

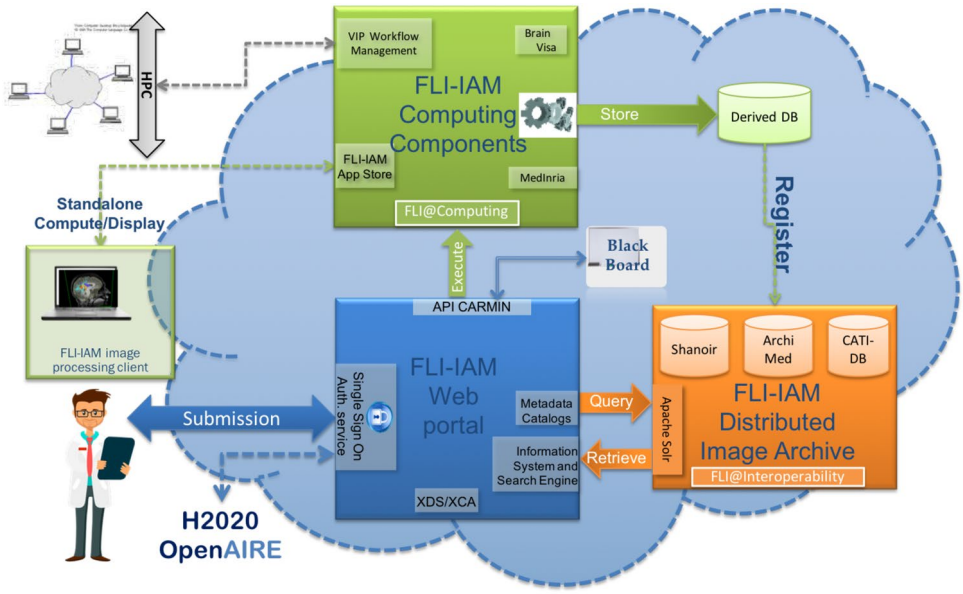

Figure 7. FLI-IAM architecture.

- anonymize and upload the training and testing data in a single place that all participants can access though the Web
- integrate and execute the image processing algorithms through a web-based portal where all algorithms are executed on the test dataset in identical conditions
- host the processed images and make them available to the participants
- provide a cloud-based integrated solution with interoperable distributed resource management systems.

Most of the past and existing challenges in the field of image processing were able to provide part of these solutions but none of them was able to provide a computing solution able to perform all of these tasks seamlessly.

We used the France Life Imaging (FLI) - Information Analysis and Management (IAM) (FLI-IAM in short) computing infrastructure for this challenge. FLI is a national infrastructure, which aims to coordinate and harmonize the network of resources on *in-vivo* imaging in France. Its IAM node represents the computing node of France Life Imaging (https://www.francelifeimaging.fr/en/about/noeuds/iam/). This architecture allows (see Fig. 7) the storage and management of preclinical and clinical *in vivo* imaging data and offers services of images processing and analysis.

FLI-IAM is based on existing, technologically ready software solutions, coming from multiple research teams over France. It proposes a web portal (blue box in Fig. 7) to unify the access to all resources and tools and provides multiple solutions for storage and computation on medical images.

To operate this challenge, three components of the entire FLI-IAM portfolio have been used:

- Web portal
- Shanoir (SHAring NeurOImaging Resources) for the database[18]
- VIP (Virtual Imaging Platform) for the computing platform[19].

In addition to these three major components, additional tools/services have been developed to provide the required level of interoperability and to finally integrate all components into one unique workflow.

The https://portal.fli-iam.irisa.fr/msseg-challengeweb portal has been used as a communication platform with all challengers. All information concerning the challenge has been distributed there, e.g. the organizational aspects, dataset descriptions, evaluation details, etc. Challengers had to subscribe on the portal to participate in the challenge.

https://shanoir-challenges.irisa.frShanoir (SHAring NeurOImaging Resources) served as central database for all datasets necessary for the challengers, all their processed results and challenger's scores. Shanoir is an open source neuro-informatics platform designed to share, archive, search and visualize neuroimaging data. It provides a user-friendly secure web access and offers an intuitive workflow to facilitate the collection and retrieval of neuroimaging data from multiple sources. Shanoir comes along many features such as anonymization of data, support for multi-center clinical studies on subjects or group of subjects.

http://vip.creatis.insa-lyon.frVIP (Virtual Imaging Platform) provided all necessary resources for the integration and the execution of all challenger processing pipelines. The pipelines were provided by challengers as https://www.docker.comDocker containers and were integrated into VIP using the Boutiques application repository. Boutiques relies on Linux containers to solve the problem of application installation in a lightweight manner and it uses a versatile JSON format to describe command line tools. VIP also ensured the execution of the challenger pipelines (and the subsequent segmentation performance analysis) on the computing resources available for the challenge.

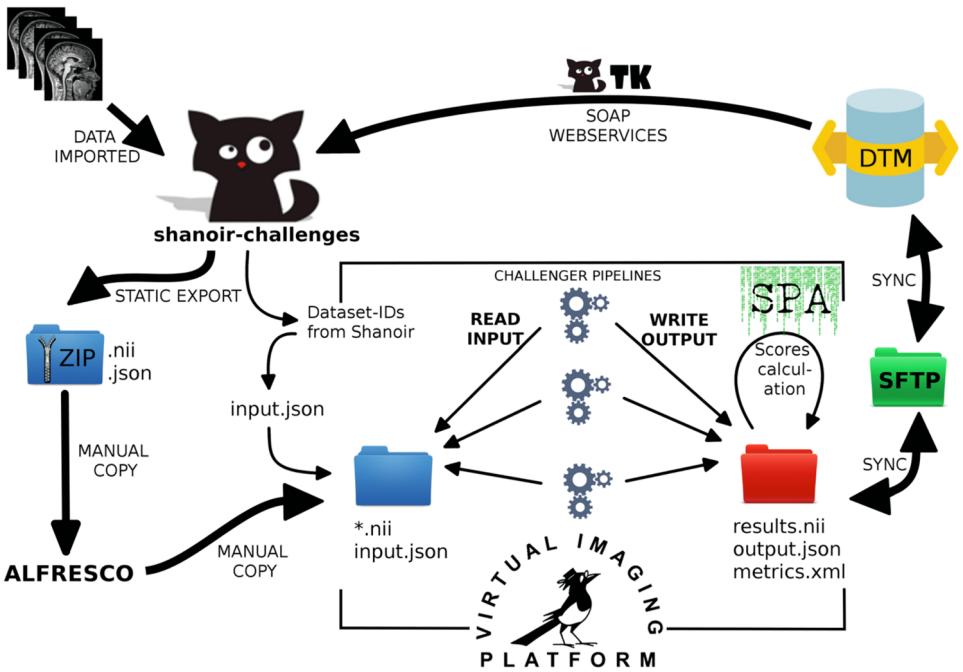

Figure 8. Workflow for database and computing platform integration.

Figure 8 gives an overview of the integration level between database and computing platform and describes the workflow that was set up for the hosting of the challenge.

After the preparation of the challenge data, all source datasets were imported into Shanoir. The training data were shared with the challengers using the portal and its file download feature. The testing data was processed by VIP using a static export folder exported from the database - containing all necessary meta-data from the database to import results back into the database and attach them to the source dataset for each challenger.

After the algorithms produced their results, the segmentation performance analysis (including all measures described in the next section) was run to compare challengers results with the consensus ground truth and calculate the scores. The continuously running DataTransferModule has been connected to the results folder of VIP to automatically import back the result datasets into Shanoir. All result datasets and their corresponding scores have been made available in Shanoir for challengers. The challenge organization team could then easily access the scores summary within Shanoir.

Challenge Evaluation Strategy and Metrics. Using the computing platform, challengers were asked to provide algorithms delineating lesions in the FLAIR reference space. This was asked to match the space in which the manual delineations were carried out and therefore avoid any unwanted discrepancies due to interpolation of the challengers' results. To evaluate these results, we have implemented a large set of evaluation measures for the challenge with the goal of evaluating the different aspects evaluated by clinicians when looking at MS patient images. For this reason, we have separated the evaluation into two major categories of evaluation metrics:

- Segmentation evaluation: does the algorithm provide a precise delineation of each lesion?
- Lesion detection evaluation: does the algorithm find all lesions in the image independently of its precise delineation?

Each of these categories may contain several metrics that characterize differently the segmentation quality. We describe in more details each of the chosen metrics for the challenge in the following sections. All evaluation algorithms used for this paper are available open-source as part of the Anima software (http://github.com/Inria-Visages/Anima-Public). Although not presented in this article (but available as part of supplementary material (https://doi.org/10.5281/zenodo.1307653), we remind the strategy that was used at the challenge to rank, for each of these metrics, the different methods on all patients:

- For each patient, compute the selected metric for each algorithm by comparing it to the ground truth previously computed
- For each patient, rank the algorithms according to the selected metric (from 1: best performing to *N*: worst performing)
- Compute for each algorithm its average ranking over all patients evaluated. This average rank is used for the final ranking of the methods.

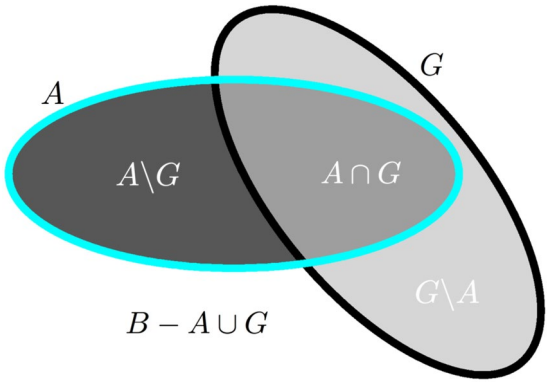

Figure 9. Illustration of overlap-based segmentation evaluation: quantities used for measures computation. A denotes the evaluated segmentation, G the ground truth, and B the image domain.

We had selected this approach instead of simply averaging the metric scores for each algorithm to avoid a bias of some methods that would get a few very good metric scores that would not represent their true behavior. This approach instead considers as the best method the one that ranks the best on average for all patients evaluated, thereby discarding this bias problem. Instead in this work, we focus more on the graphical analysis of the cluster analysis of the algorithms with respect to the experts who delineated the structures. To this end, we performed a multi-parametric analysis of the results. For each couple of metrics presented in the following (average surface distance, Dice score and F_1 score), we computed a 2D scatter representation of the average results on all testing patients of each of the teams and of the experts. Since different clusters of results quality may be outlined by such graphs, we then ran for each combination of metrics a clustering into three groups of the average performance of the teams and experts. For this clustering to be precise enough however, we need to account for the variance around the average points. We have therefore chosen to perform a spectral clustering[20], considering each point of the 2D graph not as a mean but as a multivariate Gaussian, using a distance between multivariate Gaussians as expressed in[21], thus accounting for the covariance in the individual scores.

Segmentation evaluation. The first category of evaluation metrics is also the most known in the literature and concerns segmentation evaluation, i.e. are the contours of the lesions precisely delineated compared to the ground truth. In this group, we distinguish two sub-categories, each quantifying the precision of lesions delineation: overlap-based and surface-based metrics. In the following, we will consider two binary images representing respectively the lesions consensus (i.e. the ground truth): G, and the evaluated segmentation (i.e. one algorithm segmentation result): A, both illustrated in Fig. 9.

Overlap metrics: These measures consider the voxel-based overlap of A and G based on the quantities illustrated in Fig. 9. Among those measures, we use the following ones:

- Dice score[22]: $D = 2\frac{|A \cap G|}{|A| + |G|}$
- Positive predictive value: $P = \frac{|A \cap G|}{|A|}$
- Sensitivity: $\mathrm{Se} = \frac{|A \cap G|}{|G|}$
- Specificity: $\mathrm{Sp} = \frac{|B| - |A \cup G|}{|B| - |G|}$

where $A \cup G$ is computed from other quantities: $A \cup G = A \cap G + A\backslash G + G\backslash A$. For all formulas in this section, the notation $|\cdot|$ denotes taking the cardinal of a set of voxels, e.g. $|A \cap G|$ denotes the number of voxels in that set. As a final remark for this category, the choice of the size of image B is quite important as it will influence specificity. A too large region for B could indeed lead all specificity values to be very close to 1 by construction and therefore make them difficult to compare. We therefore chose for the challenge to compute B as the union of all available segmentations for a patient (automatic and manual), dilated three times by a 6-connectivity kernel.

Each overlap-based metric varies between 0 and 1, 1 being a perfect result and 0 the worst result. Each measure is however sensitive to a different phenomenon in the quality of segmentations: positive predictive value and specificity are influenced by false positives and are therefore sensitive to overly large segmentations; sensitivity is influenced by false negatives and is thus sensitive to overly small segmentations. Finally, the Dice score is a composite measure attempting to summarize all influences into a single scalar measure.

Surface metric: In addition to overlap-based metrics, we have computed the average symmetric surface distance, also used in MICCAI 2008 challenge on MS lesions segmentation organized by[7]. Instead of using voxel-based overlaps, this measure uses contours extracted from the two input segmentations A and G, denoted respectively A_S and G_S. This distance is expressed as the following sum:

$$S = \frac{\sum_{i \in A_S} d(x_i,\ G_S) + \sum_{j \in G_S} d(x_j,\ A_S)}{N_A + N_G} \quad (1)$$

where d denotes the minimal Euclidean distance between a point of one surface and the other surface, N_A and N_G denote the number of points of each surface.

Detection evaluation. As mentioned in the introduction, evaluation of the detection of lesions is as crucial, if not even more, as segmentation precision as the number of lesions is used for MS diagnosis. We wanted to evaluate in this category how many lesions have been (in)correctly detected, independently of the precision of their contours.

Defining lesion detection: This whole category of measures relies on identifying individual lesions in the ground truth G and evaluated segmentations. For this task, we first compute the connected components of G and A (with a 18-connectivity kernel) and remove all lesions that are smaller in size than 3 mm^3. We therefore get label images $\widetilde{G}$ and $\widetilde{A}$ where each label denotes a specific lesion.

From these two labeled images, two quantities are computed that will be used to characterize the detection power of an algorithm:

- TP_G: the number of lesions among the M lesions in the ground truth $\widetilde{G}$ that are correctly detected by $\widetilde{A}$
- TP_A: the number of lesions among the N lesions in the automatic segmentation $\widetilde{A}$ that are correctly detected by $\widetilde{G}$.

Let us consider only the case of TP_G, TP_A being computed with the same procedure but reverting the roles of $\widetilde{A}$ and $\widetilde{G}$. We first construct the joint histogram H of $\widetilde{A}$ and $\widetilde{G}$ where H_{ij} corresponds to the number of voxels having label $I \in \{0M\}$ in $\widetilde{G}$ and label $j \in \{0N\}$ in $\widetilde{A}$. We consider a lesion j in $\widetilde{G}$ ($\widetilde{G}_j$) to be detected if it respects the following rules:

- The lesion $\widetilde{G}_j$ is overlapped at least at a rate of α% by lesions of $\widetilde{A}$
- Lesions of $\widetilde{A}$ that contribute the most to the detection of $\widetilde{G}_j$ (summing up to γ% of the total overlap) do not go outside of $\widetilde{G}_j$ by more than β%.

While the first condition ensures that the lesion to be detected is sufficiently overlapped, the second condition ensures that the detection is not due to an overly large segmentation in $\widetilde{A}$ that would overlap many lesions in $\widetilde{G}$ by chance. These two conditions are implemented in Algorithm 1.

Algorithm 1. TP_G computation algorithm.

Let $\mathrm{TP}_G = 0$, construct H
for $i \in \{1 \dots M\}$ **do**
  Compute $S_i : S_i = \frac{\sum_{j>0} H_{i,j}}{\sum_{j\geq 0} H_{i,j}}$
  if $S_i > \alpha \in [0,1]$ **then**
    Construct a sorting vector $p[k] : k \in \{1 \dots N\} \rightarrow \{1 \dots N\}$ so that $H_{i,p[k]}$ is sorted in decreasing order
    Let $wSum = 0, k = 0, \mathrm{vAccept} = true$
    while $wSum < \gamma \in [0,1]$ **do**
      Compute $T_k : T_k = \frac{H_{0,p[k]}}{\sum_{l\geq 0} H_{l,p[k]}}$
      if $T_k > \beta \in [0,1]$ **then**
        $\mathrm{vAccept} = false$
        Break
      end if
      $wSum = wSum + \frac{H_{i,p[k]}}{\sum_{l>0} H_{i,l}}$
      $k = k + 1$
    end while
    if vAccept is true **then**
      Let $\mathrm{TP}_G = \mathrm{TP}_G + 1$
    end if
  end if
end for

For the challenge, we used this algorithm with values heuristically defined on several independent tests to give meaningful values for TP_G and TP_A: $\alpha = 10\%$, $\gamma = 65\%$, $\beta = 70\%$.

Detection metrics: From the number of lesions M and N respectively in $\widetilde{G}$ and $\widetilde{A}$, and the numbers computed above (TP_G and TP_A), the following detection metrics are computed, named after their similarity to overlap-based metrics:

- Lesion sensitivity, i.e. the proportion of detected lesions in $\widetilde{G}$: $Se_L = \frac{TP_G}{M}$
- Lesion positive predictive value, i.e. the proportion of true positive lesions inside $\widetilde{A}$: $P_L = \frac{TP_A}{N}$.

In addition to these two metrics, we have computed a summary metric to get, like the Dice score for segmentation metrics, a one-glance idea of the detection performance of a given method (0 meaning worst performance and 1 meaning perfect detection performance). This summary metric, the F_1 score, considers both lesion sensitivity and positive predictive value to compute the score. It is defined as follows:

$$F_1 = 2\frac{Se_L P_L}{Se_L + P_L}. \tag{2}$$

References

1. Pugliatti, M. *et al*. The epidemiology of multiple sclerosis in europe. *European Journal of Neurology* 700–722 (2006).
2. Polman, C. H. *et al*. Diagnostic criteria for multiple sclerosis: 2010 Revisions to the McDonald criteria. *Annals of Neurology* **69**, 292–302 (2011).
3. Leray, E. *et al*. Evidence for a two-stage disability progression in multiple sclerosis. *Brain* **133**, 1900–1913 (2010).
4. Mortazavi, D., Kouzani, A. Z. & Soltanian-Zadeh, H. Segmentation of multiple sclerosis lesions in mr images: a review. *Neuroradiology* **54**, 299–320 (2012).
5. Lladó, X. *et al*. Segmentation of multiple sclerosis lesions in brain mri: A review of automated approaches. *Information Sciences* **186**, 164–185 (2012).
6. García-Lorenzo, D., Francis, S., Narayanan, S., Arnold, D. L. & Collins, D. L. Review of automatic segmentation methods of multiple sclerosis white matter lesions on conventional magnetic resonance imaging. *Medical Image Analysis* **17**, 1–18 (2013).
7. Styner, M. *et al*. 3D Segmentation in the Clinic: A Grand Challenge II: MS lesion segmentation. *MIDAS Journal* (2008).
8. Carass, A. *et al*. Longitudinal Multiple Sclerosis Lesion Segmentation: Resource & Challenge. *Neuroimage* **148**, 77–102 (2017).
9. Cotton, F., Kremer, S., Hannoun, S., Vukusic, S. & Dousset, V. OFSEP, a nationwide cohort of people with multiple sclerosis: Consensus minimal MRI protocol. *Journal of Neuroradiology* **42**, 133–140 (2015).
10. Akhondi-Asl, A., Hoyte, L., Lockhart, M. E. & Warfield, S. K. A Logarithmic Opinion Pool Based STAPLE Algorithm for the Fusion of Segmentations With Associated Reliability Weights. *IEEE Transactions on Medical Imaging* **33**, 1997–2009 (2014).
11. Filippi, M. *et al*. Quantitative brain mri lesion load predicts the course of clinically isolated syndromes suggestive of multiple sclerosis. *Neurology* **44**, 635–635 (1994).
12. Rudick, R. A., Lee, J.-C., Simon, J. & Fisher, E. Significance of t2 lesions in multiple sclerosis: A 13-year longitudinal study. *Annals of Neurology* **60**, 236–242 (2006).
13. Ribeiro, A. S., Nutt, D. J. & McGonigle, J. Which metrics should be used in non-linear registration evaluation? *In Medical Image Computing and Computer-Assisted Intervention – MICCAI* **2015**, 388–395 (2015).
14. Coupé, P. *et al*. An optimized blockwise nonlocal means denoising filter for 3-D magnetic resonance images. *IEEE Transactions on Medical Imaging* **27**, 425–441 (2008).
15. Commowick, O., Wiest-Daesslé, N. & Prima, S. Block-matching strategies for rigid registration of multimodal medical images. In *9th IEEE International Symposium on Biomedical Imaging (ISBI)*, 700–703 (2012).
16. Manjón, J. V. & Coupé, P. volBrain: An Online MRI Brain Volumetry System. Frontiers in. *Neuroinformatics* **10**, 30 (2016).
17. Tustison, N. J. *et al*. N4ITK: Improved N3 Bias Correction. *IEEE Transactions on Medical Imaging* **29**, 1310–1320 (2010).
18. Barillot, C. *et al*. Shanoir: Applying the Software as a Service Distribution Model to Manage Brain Imaging Research Repositories. *Frontiers in information and communication technologies* (2016).
19. Glatard, T. *et al*. A virtual imaging platform for multi-modality medical image simulation. *IEEE Transactions on Medical Imaging* **32**, 110–118 (2013).
20. Ng, A. Y., Jordan, M. I. & Weiss, Y. On spectral clustering: Analysis and an algorithm. *In Advances in Neural Information Processing Systems* **14**, 849–856 (2001).
21. Calvo, M. & Oller, J. An explicit solution of information geodesic equations for the multivariate normal model. *Statistics and Decisions* 9 (1991).
22. Dice, L. Measures of the amount of ecologic association between species. *Ecology* **26**, 297–302 (1945).
23. Beaumont, J., Commowick, O. & Barillot, C. Multiple sclerosis lesion segmentation using an automated multimodal graph cut. In *Proceedings of the 1st MICCAI Challenge on Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure - MICCAIMSSEG*, 1–7 (2016).
24. García-Lorenzo, D., Lecoeur, J., Arnold, D., Collins, D. L. & Barillot, C. Multiple sclerosis lesion segmentation using an automatic multimodal graph cuts. In *12th International Conference on Medical Image Computing and Computer Assisted Intervention*, vol. 5762 of LNCS, 584–591 (2009).
25. Beaumont, J., Commowick, O. & Barillot, C. Automatic Multiple Sclerosis lesion segmentation from Intensity-Normalized multi-channel MRI. In *Proceedings of the 1st MICCAI Challenge on Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure - MICCAI-MSSEG*, 8–15 (2016).
26. Karpate, Y., Commowick, O. & Barillot, C. Robust Detection of Multiple Sclerosis Lesions from Intensity-Normalized Multi-Channel MRI. *In SPIE Medical Imaging* (2015).
27. Forbes, F., Doyle, S., Garcia-Lorenzo, D., Barillot, C. & Dojat, M. A weighted multi-sequence markov model for brain lesion segmentation. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*, 225–232 (2010).
28. Forbes, F., Doyle, S., García-Lorenzo, D., Barillot, C. & Dojat, M. Adaptive weighted fusion of multiple MR sequences for brain lesion segmentation. In *ISBI*, 69–72 (2010).
29. Khademi, A., Venetsanopoulos, A. & Moody, A. R. Generalized method for partial volume estimation and tissue segmentation in cerebral magnetic resonance images. *Journal of Medical Imaging* **1**, 14002 (2014).
30. Knight, J. & Khademi, A. MS Lesion Segmentation Using FLAIR MRI Only. In *Proceedings of the 1st MICCAI Challenge on Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure - MICCAI-MSSEG*, 21–28 (2016).
31. Mahbod, A., Chowdhury, M., Smedby, Ö. & Wang, C. Automatic brain segmentation using artificial neural networks with shape context. *Pattern Recognition Letters* **101**, 74–79 (2018).
32. Mahbod, A., Wang, C. & Smedby, Ö. Automatic multiple sclerosis lesion segmentation using hybrid artificial neural networks. In *Proceedings of the 1st MICCAI Challenge on Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure - MICCAIMSSEG*, 29–36 (2016).

33. McKinley, R. *et al.* Nabla-net: a deep dag-like convolutional architecture for biomedical image segmentation: application to white-matter lesion segmentation in multiple sclerosis. In *Proceedings of the 1st MICCAI Challenge on Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure - MICCAI-MSSEG*, 37–43 (2016).
34. Muschelli, J., Sweeney, E., Maronge, J. & Crainiceanu, C. Prediction of MS Lesions using Random Forests. In *Proceedings of the 1st MICCAI Challenge on Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure - MICCAI-MSSEG*, 45–50 (2016).
35. Cabezas, M. *et al.* Automatic multiple sclerosis lesion detection in brain MRI by FLAIR thresholding. *Computer Methods and Programs in Biomedicine* **115**, 147–161 (2014).
36. Roura, E. *et al.* A toolbox for multiple sclerosis lesion segmentation. *Neuroradiology* **57**, 1031–1043 (2015).
37. Santos, M. M., Diniz, P. R. B., Silva-Filho, A. G. & Santos, W. P. *Evaluation-Oriented Training via Surrogate Metrics for Multiple Sclerosis Segmentation,* vol. 9901 of LNCS, 398–405 (Springer, 2016).
38. Santos, M. M., Diniz, P. R., Silva-Filho, A. G. & Santos, W. P. Evaluation- Oriented Training Strategy on MS Segmentation Challenge 2016. In *Proceedings of the 1st MICCAI Challenge on Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure- MICCAI-MSSEG*, 57–62 (2016).
39. Tomas-Fernandez, X. & Warfield, S. K. A Model of Population and Subject (MOPS) Intensities With Application to Multiple Sclerosis Lesion Segmentation. *IEEE Transactions on Medical Imaging* **34**, 1349–1361 (2015).
40. Tomas-Fernandez, X. & Warfield, S. K. MRI Robust Brain Tissue Segmentation with application to Multiple Sclerosis. In *Proceedings of the 1st MICCAI Challenge on Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure - MICCAIMSSEG*, 63–67 (2016).
41. Urien, H., Buvat, I., Rougon, N., Soussan, M. & Bloch, I. Brain lesion detection in 3D PET images using max-trees and a new spatial context criterion. In *International Symposium on Mathematical Morphology (ISMM)*, vol. 10225 of LNCS, 455–466 (2017).
42. Urien, H., Buvat, I., Rougon, N. & Bloch, I. A 3D hierarchical multimodal detection and segmentation method for multiple sclerosis lesions in MRI. In *Proceedings of the 1st MICCAI Challenge on Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure - MICCAI-MSSEG*, 69–73 (2016).
43. Valverde, S. *et al.* Improving automated multiple sclerosis lesion segmentation with a cascaded 3D convolutional neural network approach. *Neuroimage* **155**, 159–168 (2017).
44. Vera-Olmos, F., Melero, H. & Malpica, N. Random Forest for Multiple Sclerosis Lesion Segmentation. In *Proceedings of the 1st MICCAI Challenge on Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure - MICCAI-MSSEG*, 81–86 (2016).

Acknowledgements

This work was partly funded by France Life Imaging (grant ANR-11-INBS-0006 from the French "Investissements d'Avenir" program) for funding and sponsoring the challenge. This work has also been partly supported by a grant (OFSEP) provided by the French State and handled by the "Agence nationale de la recherche", within the framework of the "Investissements d'Avenir" program, under the reference ANR-10-COHO-002. We also thank the French national cohort OFSEP (a French "Investissements d'Avenir" program), and particularly the imaging group inside this cohort consortium for their constant support, fruitful discussions on the challenge and providing the MR images.

Author Contributions

Jérémy Beaumont, Olivier Commowick, Christian Barillot, Senan Doyle, Michel Dojat, Florence Forbes, Jesse Knight, April Khademi, Amirreza Mahbod, Chunliang Wang, Richard McKinley, Franca Wagner, John Muschelli, Elizabeth Sweeney, Eloy Roura, Xavier Lladó, Michel M. Santos, Wellington P. Santos, Abel G. Silva-Filho, Xavier Tomas-Fernandez, Simon K. Warfield, Héléne Urien, Isabelle Bloch, Sergi Valverde, Mariano Cabezas, Francisco Javier Vera-Olmos and Norberto Malpica designed the challengers' repsective algorithms, participated to the challenge, participated in the writing and proof-reading of the evaluated teams description in particular, and of the proof-reading of the whole article. Michaël Kain, Baptiste Laurent, Florent Leray, Mathieu Simon, Sorina Camarasu Pop, Pascal Girard, Frédéric Cervenansky, Tristan Glatard, Olivier Commowick, Christian Barillot and Michel Dojat participated in the setup of the platform and running of the experiments on the France Life Imaging platform, the writing and proof-reading of the pipeline processing description and results in particular, and of the proof-reading of the whole article. Olivier Commowick, Audrey Istace, Florent Leray, Baptiste Laurent, Roxana Améli, Jean-Christophe Ferré, Anne Kerbrat, Thomas Tourdias, Sandra Vukusic, Gilles Edan and François Cotton participated in the constitution of the evaluation database (selection of the patients, expert guidance on the delineation of the lesions), in the analysis of results, and in the writing of the corresponding sections of the paper in particular. They also participated in the proof-reading of the whole article. Olivier Commowick, Charles Guttmann, Frédéric Cervenansky, Martin Styner, Simon K. Warfield, François Cotton and Christian Barillot participated in all aspects of the challenge organization, design of the results evaluation experiments and metrics, writing and proof-reading of all sections of the paper.

Additional Information

Competing Interests: The authors declare no competing interests.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

APÊNDICE D – ARTIGO NA REVISTA *NEUROCOMPUTING*

Reprodução do Registro Bibliográfico para Identificação

SANTOS, M. M. dos; FILHO, A. G. da S.; SANTOS, W. P. dos. Deep convolutional extreme learning machines: Filters combination and error model validation. *Neurocomputing*, Elsevier, v. 329, p. 359–369, 2019.

Nota sobre Direitos Autorais

Os direitos de publicação e distribuição do artigo publicado na revista *Neurocomputing* foram transferidos para a editora *Elsevier* como parte do contrato de publicação. A versão final do manuscrito aceito poderia ser incluída nesta tese e, assim, aparecer em um repositório institucional de acesso aberto. No entanto, um período de embargo de 24 meses teria que ser cumprido. Para não adiar a publicação desta tese, não reproduzimos aqui o artigo publicado anteriormente na revista *Neurocomputing*. A versão publicada do artigo está disponível em <https://doi.org/10.1016/j.neucom.2018.10.063>.

Copyright Notice

The publication and distribution rights over the article published in Neurocomputing journal were transferred to Elsevier publisher as part of the publishing contract. The final version of the accepted manuscript could be included in this thesis and thus appear in an open acess institutional repository. However, an embargo period of 24 months would have to be met. In order not to postpone the publication of this thesis, we do not reproduce here the article previously published in the Neurocomputing journal. The published version of the article is available at <https://doi.org/10.1016/j.neucom.2018.10.063>