Journal of Electronic Imaging

JElectronicImaging.org

Aluminum alloy microstructural segmentation in micrograph with hierarchical parameter transfer learning method

Dali Chen
Pengyuan Zhang
Shixin Liu
Yangquan Chen
Wei Zhao

Dali Chen, Pengyuan Zhang, Shixin Liu, Yangquan Chen, Wei Zhao, "Aluminum alloy microstructural segmentation in micrograph with hierarchical parameter transfer learning method," *J. Electron. Imaging* **28**(5), 053018 (2019), doi: 10.1117/1.JEI.28.5.053018.

Journal of Electronic Imaging 28(5), 053018 (Sep/Oct 2019)

Aluminum alloy microstructural segmentation in micrograph with hierarchical parameter transfer learning method

Dali Chen,[a,b,*] Pengyuan Zhang,[a,b] Shixin Liu,[a,b] Yangquan Chen,[c] and Wei Zhao[d]
[a]Northeastern University, State Key Laboratory of Synthetical Automation for Process Industries, Shenyang, China
[b]Northeastern University, College of Information Science and Engineering, Shenyang, China
[c]University of California, School of Engineering, Merced, California, United States
[d]Shandong Nanshan Aluminum Industry Co. Ltd., Yantai, China

Abstract. The properties of aluminum alloy highly depend on the distribution, shape, and size of the microstructures. Thus accurate segmentation of these microstructures is crucial in the fields of material science. However, it is often challenging due to large variations in microstructural appearance and insufficiency in hand-labeled data. To address these challenges, we propose a hierarchical parameter transfer learning method for the automatic segmentation of microstructures in aluminum alloy micrograph, which can be seen as the generalization of the typical parameter transfer method. In the proposed method, we use the multilayer structure, multinetwork structure, and retraining technology. It can make full use of the advantages of different networks and transfer network parameters in the order from high transferability to low transferability. Several experiments are presented to verify the effectiveness of the proposed method. Our method achieves 98.88% segmentation accuracy and outperforms four typical segmentation methods. [DOI: 10.1117/1.JEI.28.5.053018]

Keywords: deep learning; transfer learning; microstructural segmentation; aluminum alloy; fully convolutional network.

Paper 190507 received May 28, 2019; accepted for publication Sep. 19, 2019; published online Oct. 14, 2019.

1 Introduction

Aluminum alloy is one of the most important and manifold class of materials in aerospace manufacturing because of its light weight and corrosion resistance.[1] The properties of aluminum alloy highly depend on the distribution, shape, and size of the microstructures.[2] Thus correct segmentation of these microstructures is crucial. However, the segmentation of complex microstructures is difficult and typically requires material science experts to manually divide microstructures into separate parts. This process is often slow, labor intensive, and suffers from poor repeatability.

The metal micrograph is a digital image taken through a microscope, which contains extensive details of microstructures and provides an important tool for the analysis of metal microstructures.[3] Recently, automatic metal microstructural segmentation in micrograph has attracted the attention of many scholars, which can greatly improve the efficiency of metallographic analysis task. So far, the automatic metal microstructural segmentation methods in micrograph can be mainly divided into two groups: rule-based methods and learning-based ones.

Rule-based methods segment the microstructure using presumed rules for microstructure. For example, the watershed-based method utilizes the edge information to segment microstructures and the region adjacency graph algorithm is used to reduce over-segmentation errors.[4] The mean shift algorithm, introduced in Ref. 5, fuses the watershed algorithm and region information for the extraction of grain boundary in metal micrograph. Morphology algorithms are used to explore grain boundary shape features such as piecewise linearity and connectivity.[6,7] The Markov random field-based segmentation algorithm[8] and multiscale hierarchical-based segmentation algorithm,[9] respectively, utilize the texture features to identify the different regions in micrograph. The fuzzy C-means-based method takes full advantage of the local spatial intensity feature to segment ductile iron microstructure.[10]

Learning-based methods do not need the presumed rules. Instead, they turn the segmentation problem into a classification task. Classifiers are trained from a set of microstructural features along with labels. The usual features include pixel intensities, edges feature, texture feature, and bag of visual feature, which are fed to different classifiers such as multilayer perceptron,[11] random forest,[12] optimum-path forest,[13] neural network,[14] and support vector machine (SVM).[15,16] These methods, by introducing discriminative features in microstructural representation, often outperform traditional rule-based methods. However, it is very difficult to acquire the discriminable features of microstructure generally. When the discriminable features are not known, these segmentation methods are not able to identify the different regions accurately in micrograph. This limits the general applicability of the metal microstructural segmentation techniques that operate on micrograph.

Deep learning methods have dramatically improved the conventional machine learning techniques, due to their strong ability to learn the hierarchical latent features of high-dimensional data.[17,18] These methods have been successfully applied in image processing,[19–24] speech recognition,[25–29] and many other domains such as medical science[30–32] and material science.[33] In the field of metallography, several deep learning-based methods have been proposed for the metal microstructural segmentation in micrograph. Chowdhury

*Address all correspondence to Dali Chen, E-mail: chendali@ise.neu.edu.cn

1017-9909/2019/$28.00

et al.[34] utilized the powerful feature learning ability of convolutional neural network (CNN) to obtain the microstructural feature of alloys of varying Sn–Ag–Cu compositions. Lubbers et al.[35] fused the CNN feature and manifold learning for the extraction of a low-dimensional microstructural representation. Azimi et al.[36] proposed a fully convolutional neural network (FCNN) accompanied by max-voting scheme to segment some given microstructures of low carbon steel. Ma et al.[37] proposed a deep learning-based image segmentation method for Al-La alloy microscopic images and point out that deep learning methods achieve more accurate segmentation results than the image processing methods.

In this paper, we focus our attention on the aluminum alloy microstructural segmentation in micrograph. The microstructure of aluminum alloy has different appearances, influenced by different factors such as alloying elements, rolling setup, cooling rates, heat treatment, and further post-treatments. This makes the design of rules and features more difficult, so the rule-based method and conventional machine-learning method are difficult to achieve this accurate segmentation task. The deep learning-based methods can be used to deal with these problems. However, these existing methods always need a large number of hand-labeled data to achieve accurate microstructural segmentation. This hand-labeled work of micrograph is difficult, slow, and typically requires material science experts to manually assign each micrograph pixel an accurate label.[38] In order to deal with this problem, a hierarchical parameter transfer learning (HPTL)-based microstructural segmentation method is proposed for the aluminum alloy micrograph in this paper.

Transfer learning focuses on utilizing the knowledge gained while solving an old problem to solve a different but related new problem.[39] For example, the knowledge gained while learning to segment natural images could apply when trying to segment the microstructure in micrograph. So far, the transfer learning methods have widely been applied in the different research fields,[40,41] such as computer vision[42–46] and natural language processing.[47,48] The detailed surveys of past research on transfer learning can be found in Refs. 40 and 41. Yosinski et al.[42] studied the relationship between network structure and transferability, and point out which features are transferable in deep networks and which type of networks are more suitable for transfer. Inspired by these excellent works, we propose an HPTL method that fuses the deep learning network and hierarchical parameter transfer techniques for the segmentation of microstructure in a metal micrograph. Our proposed method can be seen as the generalization of the typical parameter transfer method. Compared with the typical parameter transfer learning method, our proposed method has three contributions.

(1) *Multilayer structure*. Our proposed hierarchical method uses a new multilayer structure, and the typical method generally uses a single-layer structure. Our method consists of four layers. In the first layer, we use public datasets to train the network, and then obtain general features. Network parameters are transferred from a low layer to a high layer. This improvement can transfer network parameters in the order from high transferability to low transferability, which makes the transfer more detailed and reasonable, and is helpful for improving the effect of transfer.

(2) *Multinetwork structure*. Our proposed hierarchical method uses a multinetwork structure, and the typical method generally uses one network structure. We use the four different deep convolution neural networks (DCNNs), including VGG-19, FCN-32s, FCN-16s, and FCN-8s. These networks contain the same local structure, and the parameters are transferred between these local structures. This improvement can make full use of the advantages of different networks and help to improve network performance.

(3) *Retraining technology*. In the typical parameter transfer learning method, the transferred parameters generally are fixed in the process of network training. In our method, the transferred parameters can still be further optimized in the training process. This improvement can enable the network to better complete the task of the target domain on the basis of making full use of the public dataset.

This paper is organized as follows: Section 1 introduces prior work, focusing on the main problems with existing methods that are addressed by our model. The related work is introduced in Sec. 2. In Sec. 3, the HPTL method is proposed and its implementation details are described. Section 4 presents the experiment results on the transferability and segmentation performance. The paper is concluded in Sec. 5.

2 Related Work

2.1 *Visual Geometry Group Network*

Visual geometry group (VGG) network was first proposed by Simonyan and Zisserman,[49] and it is the dominant approach for almost all classification and segmentation tasks. The VGG network attracts increasing attentions from researchers in recent years. A typical VGG network consists of convolution layer, pooling layer, fully connected layer, ReLU layer, and SoftMax layer. These layers are connected repeatedly in a certain order.

In convolution layer, the input data r is convolved with a linear convolution filter θ, which is described by

$$h_i = \theta_i \otimes r + a_i, \tag{1}$$

where h_i represents the i'th output of convolution layer, which is also called feature map. The filter θ_i and bias a_i are trainable parameters corresponding to the i'th feature map.

In pooling layer, the input feature map is divided into a set of nonoverlapping rectangles. For each such subregion, the maximum is output.

In fully connected layer, the neurons z have connections to all activations l in the previous layer, which is described by

$$z_i = \alpha_i^{\mathrm{T}} l + b_i, \tag{2}$$

where z_i represents the i'th output neuron and the weight α_i and bias b_i are trainable parameters corresponding to neuron z_i.

In ReLU layer, a nonlinear activation function $f(x) = \max\{x, 0\}$ is used for removing the negative values from the activation map by setting them to zero.

In the SoftMax layer, the SoftMax classifier is applied to predict a single class of k mutually exclusive classes, which is described by

$$y_k = P(c = k|x, w) = \frac{\exp(x_k)}{\sum_j \exp(x_j)}, \tag{3}$$

where x_k represents the k'th input; y_k represents the k'th output, which is the probability of the k'th class; the w represents the trainable network parameters, which can be optimized by minimizing the cross-entropy error function, as follows:

$$\min\ E(w) = -\ln\ P(T|w) = -\sum_{n=1}^{N}\sum_{k=1}^{K} t_{nk}\ \ln\ y_{nk}, \tag{4}$$

where T is an $N \times K$ matrix of target variables with elements t_{nk}, and y_{nk} represents the probability that the n'th input data belongs to the k'th class, which is computed by Eq. (4).

This optimization problem can be solved by mini-batch gradient descent algorithm, which is based on the back-propagation process.

In our paper, we use the VGG-19 network architecture, which consists of sixteen convolutional, five pooling, and three fully connected layers. The 3×3 convolution kernel is applied which is with less parameters, but with the same support. This network is able to achieve empirically better performance than GoogLeNet and AlexNet.

2.2 *Fully Convolutional Neural Network*

FCNN was first proposed by Long et al.,[50] which takes input of arbitrary size and produces correspondingly sized output with efficient inference and learning. This network has been the dominant approach in semantic segmentation. The typical FCNN consists of the typical classification network (AlexNet, VGG, or GoogLeNet) and upsampling layer. In the upsampling layer, the simple bilinear interpolation technology is applied to convert the classification network into a fully convolutional network that produces a coarse output map. At the end of network, the SoftMax classifier is applied to predict the class of each pixel.

In our paper, we use the FCN-8s/16s/32s network architecture, which consists of sixteen convolutional layers, five pooling layers, three fully convolutional layers, and 3/2/1 deconvolutional layers.

3 Methods and Implementation

3.1 *Problem Description*

The microstructural segmentation in micrograph is the process of assigning each pixel a label designating the microstructural class to which it belongs (e.g., Mg_2Si or aluminum). Figure 1 shows an aluminum alloy micrograph, which contains the Mg_2Si, aluminum, and Fe-containing phase microstructures. Let x represent a given micrograph and x_n represent the n'th pixel in micrograph. Associated with the output of pixel x_n is a binary class label $t_n^k \in \{0,1\}$, where $k = 1, \ldots, K$.

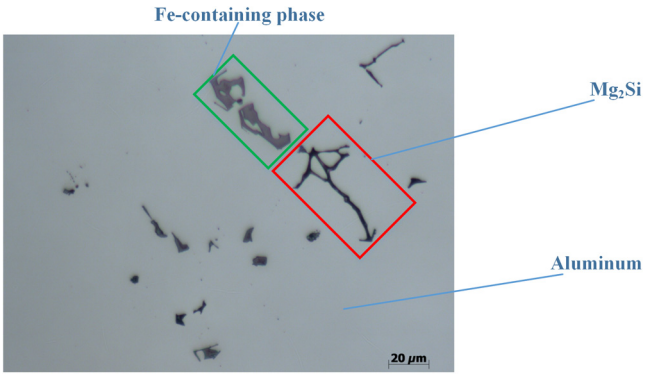

Fig. 1 Aluminum alloy micrograph.

Therefore, the posterior probability that the input pixel belongs to the k'th class is given by

$$p(t_n^k|x_n, w) = y_k(x_n, w) = \frac{\exp[f(x_n, w_k)]}{\sum_j \exp[f(x_n, w_j)]}. \tag{5}$$

Given a data set of N independent, identically distributed observations $X = \{x_1, x_2, \ldots, x_N\}$, along with corresponding target values $T = \{t_1, t_2, \ldots, t_N\}$. Assume that the class labels are independent, then the conditional distribution of the targets is given by

$$p(t_n|x_n, w) = \prod_{k=1}^{K} y_k(x_n, w)^{t_n^k}[1 - y_k(x_n, w)]^{1-t_n^k}. \tag{6}$$

In this segmentation problem, each pixel is assigned to one of K mutually exclusive classes. So the negative logarithm of the corresponding likelihood function then gives the following error function:

$$E(w) = -\ln\ P(T|X, w) = -\sum_{n=1}^{N}\sum_{k=1}^{K} t_n^k\ \ln\ y_k(x_n, w), \tag{7}$$

and the parameter w can be optimized by minimizing the error function $E(w)$.

When $f(x_n, w_k) = w_k^{\mathrm{T}} x_n$ and x_n is the feature vector, this segmentation model is equal to the logistic regression that is a typical model for classification. Unfortunately, it is always difficult to obtain proper feature vectors. In order to deal with this problem, we use a popular FCNN to model $f(x_n, w_k)$, which has strong ability to learn the hierarchical latent features of high-dimensional data. The network architecture of FCN is shown in Fig. 2.

This optimization problem min $E(w)$ can be solved by mini-batch gradient descent algorithm, which is based on the back-propagation process.

3.2 *Hierarchical Parameter Transfer Technique*

FCNN is the dominant approach in semantic segmentation and has a strong ability to provide accurate segmentation. However, in FCNN, we need a large amount of hand-labeled data to achieve accurate microstructural segmentation. This hand-labeled micrograph work is difficult, slow, and typically requires materials science experts to manually assign

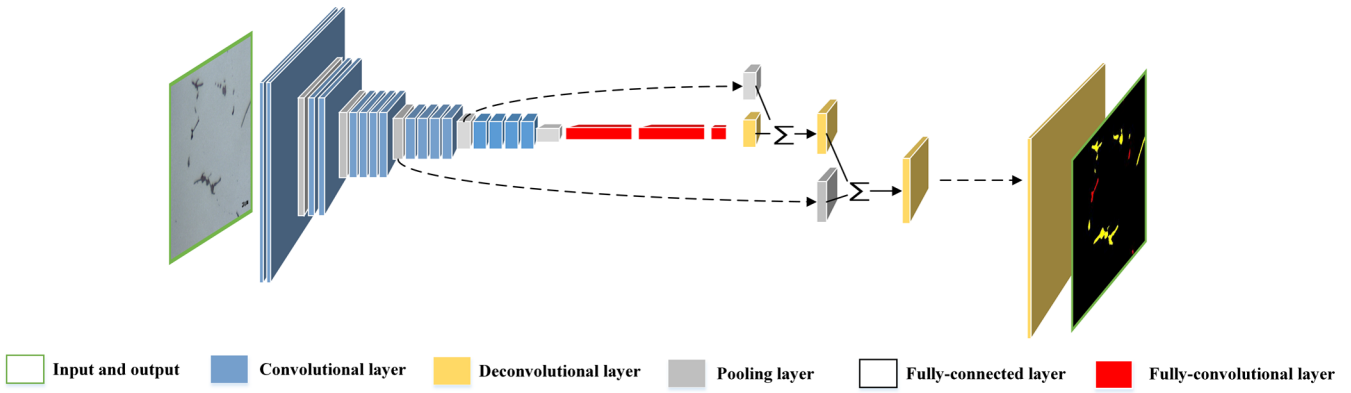

Fig. 2 Network architecture of FCN.

each micrograph pixel an accurate label. For this problem, a hierarchical parameter transfer technique is proposed, which fuses the typical VGG network, FCN-32s, FCN-16s, and FCN-8s to achieve small-sample learning.

This proposed hierarchical parameter transfer technique is inspired by the work in Ref. 42. This work studies the relationship between general features and specific features, and points out that the general features are transferable in deep networks. Therefore, we define a set of transferable general features between each deep network, which is used to transfer the knowledge between each deep network indirectly and deals with the problem of imprecise segmentation caused by the shortage of the samples with accurate labels. For easy description, the detailed process of the proposed hierarchical parameter transfer technique is shown in Fig. 3.

As shown in Fig. 3, the proposed hierarchical parameter transfer technique includes three main steps. In the first step, the VGG-19 network is trained using famous ImageNet dataset, and we define that the first two layers (four convolutional and two pooling layers) features are general and transferable, as shown in the first layer in Fig. 3. These general features are transferred to the FCN-32s network. In the second step, the FCN-32s network is trained using given micrograph dataset, and we define that the first eight layers (sixteen convolutional, five pooling, and three fully convolutional layers) features are general and transferable, as shown in the second layer in Fig. 3. These general features are transferred to the FCN-16s network. Similarly, in the third step, the FCN-16s network is trained using given micrograph dataset, and the first nine layers (sixteen convolutional, five pooling, three fully convolutional, and one deconvolutional layers) features are transferred to the FCN-8s network, as shown in the third layer in Fig. 3.

In this process, a large number of ImageNet data with accurate labels are used to learn the general features of network, which alleviates the problem caused by the shortage of the hand-labeled samples. Moreover, the proposed hierarchical parameter transfer technique is the generalization of the typical parameter transfer technique. When there is only one step, the proposed hierarchical parameter transfer method is the same as the typical parameter transfer method. A large number of experimental results show that the reasonable hierarchical strategy can improve the prediction performance of deep learning network.

3.3 *Implementation Details*

All models are trained and tested with TensorFlow on a single NVIDIA Titan XP.

3.3.1 *Training and test dataset*

The problem of insufficient training data is inevitable in practice for the microstructural segmentation in micrograph. In order to deal with this problem, we use a subset of the large hand-labeled ImageNet dataset as the training dataset in source domain D_S, which includes about 10 million labeled images depicting 1000 object categories. In the target domain, the 50 aluminum alloy micrographs accurately labeled by an expert are used as training dataset D_T. The data augmentation technique is used in the target domain training dataset D_T, which increases the number of data to 1500 micrographs. The test dataset D_V in target domain includes 40 aluminum alloy micrographs. These aluminum alloy micrographs are taken through the Axio Imager Observer and has the size of 2560×1920 pixels.

3.3.2 *Network architecture*

In our paper, we use four typical deep learning networks including VGG-19, FCN-32s, FCN-16s, and FCN-8s. The VGG-19 network architecture includes sixteen convolutional, five pooling, and three fully connected layers. The FCN network combines the typical classification network with FCNs and accomplishes the semantic segmentation task. The skip architecture that combines semantic information with appearance information is first applied in FCN network. In this paper, we use the FCN-8s/16s/32s network architecture, which includes sixteen convolutional, five pooling, three fully convolutional, and 3/2/1 deconvolutional layers.

3.3.3 *Transferable parameter*

In our proposed deep transfer learning network, it is vital to determine which layer is transferable. We define the transferable layer in VGG-19, FCN-32s, FCN-16s, and FCN-8s, respectively. The transferable layers between VGG-19 and FCN-32s are the first two layers, including four convolutional and two pooling layers. For easy description, the parameter of these transferable layers are denoted as ω_{t1}. The transferable layers between FCN-32s and FCN-16s are the first eight layers, including sixteen convolutional, five pooling, and three fully convolutional layers. For convenient

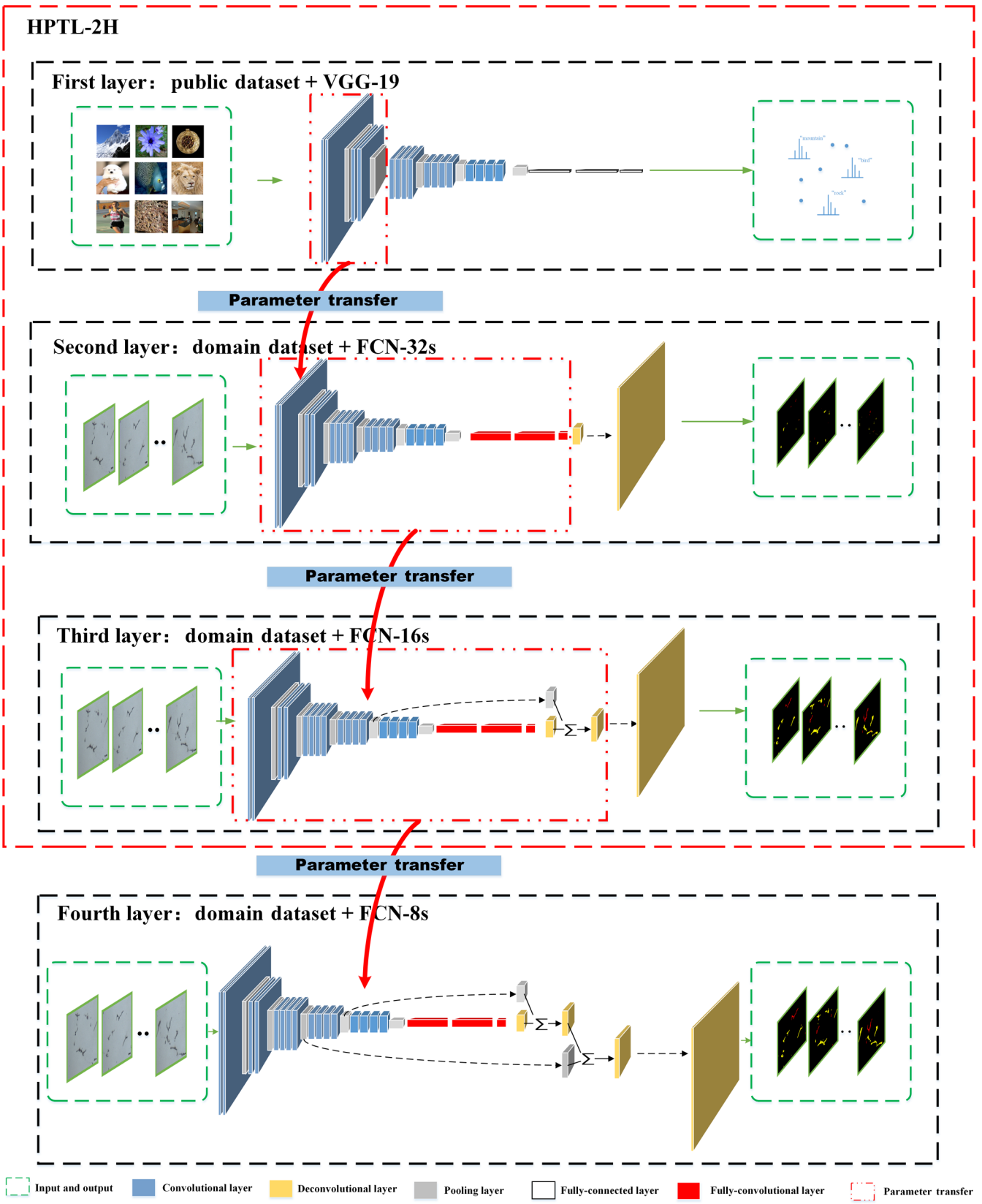

Fig. 3 Workflow of the proposed HPTL method.

description, the parameters of these transferable layers are denoted as ω_{t2}. The transferable layers between FCN-16s and FCN-8s are the first nine layers, including sixteen convolutional, five pooling, three fully convolutional, and one deconvolutional layers. For easy description, the parameters of these transferable layers are denoted as ω_{t3}. In Sec. 5, we show the appearance of some feature maps from general to specific in our network.

3.3.4 *Parameter description*

There are many parameters in our deep transfer learning network. For convenient description of VGG-19 network, we shall denote the number of classification by $c_1 = 1000$, the number of layer by $o_1 = 24$, and the network parameters by ω_1. In the process of training VGG-19 network, we use a fixed learning rate γ_1 of 0.001, a momentum Ψ_1 of 0.9, and weight decay τ_1 of 0.95 in stochastic gradient descent (SGD) algorithm. The training process stops when the iteration number $\kappa_1 = 120$k. Similarly, for FCN-8s/16s/32s networks, we shall denote the number of classification by $c_2 = 3$ including (Mg_2Si, aluminum, and Fe-containing phase), the number of layer by $o_{8s} = 27/o_{16s} = 26/o_{32s} = 25$ and the network parameters by $\omega_{8s}/\omega_{16s}/\omega_{32s}$. In the process of training FCN network, we use the learning rate γ_2 of 0.0001 to train FCN-32s, FCN-16s, and FCN-8s. The input images are cropped to 224×224 pixels size with batch size of 16. The Adam (adaptive moment estimation) algorithm is used to optimize the network parameters ω_{8s}, ω_{16s}, and ω_{32s}. For the moment estimation, the exponential decay rates $\tau_2 = 0.9$ and $\tau_3 = 0.999$. The training process stops when the iteration number $\kappa_2 = 12$k. Here we use random initializations for the weights ω_1, ω_{8s}, ω_{16s}, and ω_{32s}. Dropout technique is used in VGG-19 and FCN networks in order to improve generalization power of network by randomly dropping neurons from the network architecture during training phase. Moreover, the L_2 regularization is used in FCN-8s/16s/32s networks, which is implemented by penalizing the squared magnitude of all parameters directly in the objective. The regularization coefficient $\lambda = 0.5$. For convenient reading, the description of parameters in our paper is summarized in Table 1.

To summarize, our entire microstructural segmentation algorithm in a form of a pseudocode is done as follows.

(1) Initialization:

- Training dataset in source domain D_S;
- Training dataset in target domain D_T;
- *Random initialization*: Network parameters ω_1, ω_{8s}, ω_{16s}, and ω_{32s}; transferable parameters ω_{t1}, ω_{t2}, and ω_{t3};
- For VGG-19 network, $c_1 = 1000$, $o_1 = 24$, $\gamma_1 = 0.01$, $\Psi_1 = 0.9$, $\tau_1 = 0.95$, and $\kappa_1 = 120$k;
- For FCN-8s/16s/32s network, $c_2 = 3$, $o_{8s} = 27$, $o_{16s} = 26$, $o_{32s} = 25$, $\gamma_2 = 0.0001$, $\tau_2 = 0.9$, $\tau_3 = 0.999$, $\kappa_2 = 12$k, and $\lambda = 0.5$.

(2) *Training*: Compute ω_1, ω_{8s}, ω_{16s}, ω_{32s}, ω_{t1}, ω_{t2}, and ω_{t3} by the following steps.

- Step 1: Optimize ω_1 by SGD algorithm based on dataset D_S.
- Step 2: Transfer ω_{t1} to FCN-32s and compute ω_{32s} by Adam algorithm based on dataset D_T.
- Step 3: Transfer ω_{t2} to FCN-16s and compute ω_{16s} by Adam algorithm based on dataset D_T.
- Step 4: Transfer ω_{t3} to FCN-8s and compute ω_{8s} by Adam algorithm based on dataset D_T.
- Step 5: Output ω_{8s}.

(3) *Prediction*: Segment the given micrograph x by FCN-8s with parameter ω_{8s}.

Table 1 Parameter description.

Symbol	Description
D_S	Training dataset in source domain
D_T	Training dataset in target domain
D_V	Test dataset in target domain
ω_1	VGG-19 network parameters
ω_{8s}	FCN-8s network parameters
ω_{16s}	FCN-16s network parameters
ω_{32s}	FCN-32s network parameters
ω_{t1}	Transferable parameters between VGG-19 and FCN-32s
ω_{t2}	Transferable parameters between FCN-32s and FCN-16s
ω_{t3}	Transferable parameters between FCN-16s and FCN-8s
c_1	Number of classification in D_S
c_2	Number of classification in D_T
o_1	Number of layer in VGG-19
o_{8s}	Number of layer in FCN-8s
o_{16s}	Number of layer in FCN-16s
o_{32s}	Number of layer in FCN-32s
γ_1	Learning rate of VGG-19
γ_2	Learning rate of FCN-8s/16s/32s
τ_1	Weight decay of VGG-19
τ_2	Exponential decay rate of FCN-8s/16s/32s
τ_3	Exponential decay rate of FCN-8s/16s/32s
κ_1	Iteration number of VGG-19
κ_2	Iteration number of FCN-8s/16s/32s
λ	Regularization coefficient

4 Experiments and Analysis

In order to verify the effectiveness of our proposed algorithm, a large number of experiments are performed for the analysis of transferability and segmentation performance.

4.1 *Analysis of Transferability*

The aim of this experiment is to analyze the transferability of the proposed HPTL method. The FCN-8s is taken as an example in this experiment. For easy description, we divide the FCN-8s network into five parts with no overlap, including conv1 (two convolutional and one pooling layers), conv2 (two convolutional and one pooling layers), conv3 (four convolutional and one pooling layers), conv4 (four

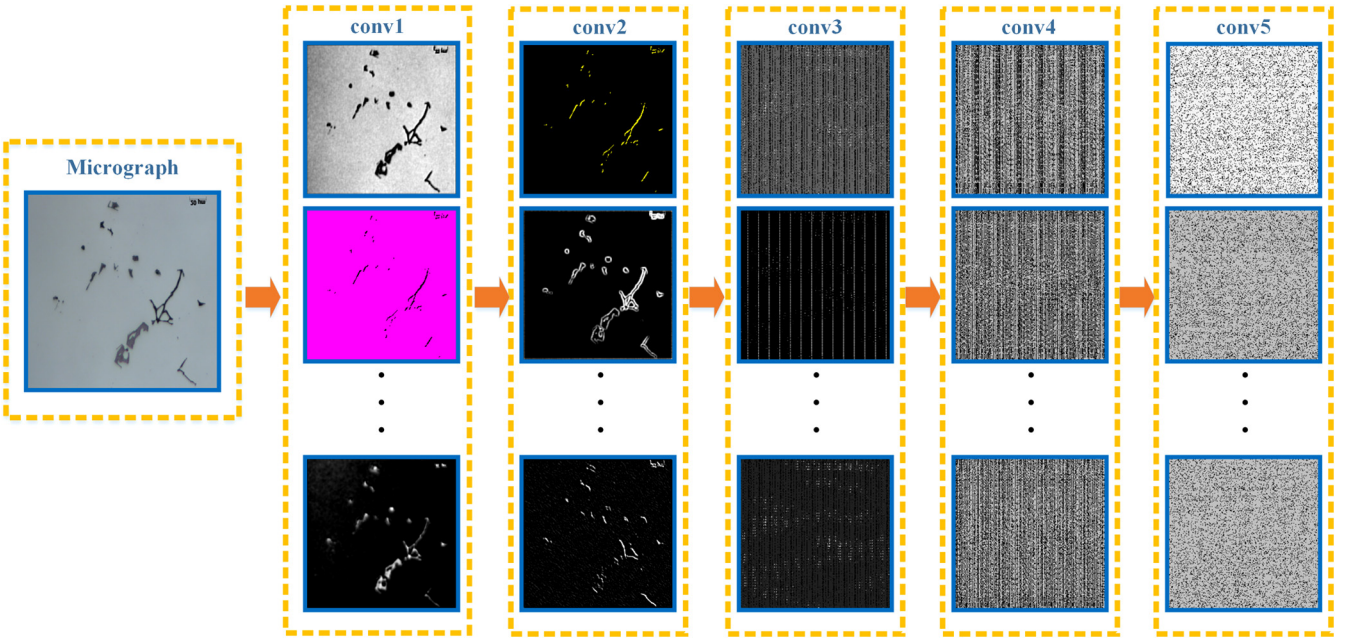

Fig. 4 Feature maps of aluminum alloy micrograph.

convolutional and one pooling layers), and conv5 (four convolutional and one pooling layers). Some representative feature maps in different parts are chosen and shown in Fig. 4. As shown in Fig. 4, the representative feature maps of different parts are shown in the different columns. For example, three feature maps of conv1 are shown in the first column. From Fig. 4, we can see that the feature maps of conv1 and conv2 are the general feature of microstructure, including shape, color, and edge. Conversely, the feature maps of conv3, con4, and conv5 are specific. These experiment results verify the rationality of our HPTL method.

4.2 *Analysis of Segmentation Performance*

The aim of this section is to analyze the segmentation performance of the proposed HPTL method. For the performance measure of HPTL, we use the accuracy (acc), which has been largely used in literature and commonly applied to determine the quality of a processed image. It can be calculated by the following equation:

$$\mathrm{acc}(f_{\mathrm{HPTL}}; D_v) = \frac{1}{N}\sum_{i=1}^{N}\prod[f_{\mathrm{HPTL}}(x_i) = t_i], \tag{8}$$

where D_v is the test dataset in target domain, which includes 40 aluminum alloy micrographs. $\prod[f_{\mathrm{HPTL}}(x_i) = t_i]$ is the indicator function defined as

$$\prod[f_{\mathrm{HPTL}}(x_i) = t_i] = \begin{cases} 1, & f_{\mathrm{HPTL}}(x_i) = t_i \\ 0, & f_{\mathrm{HPTL}}(x_i) \neq t_i \end{cases}. \tag{9}$$

Here t_i is the true label of the given x_i and f_{HPTL} denotes our segmentation method.

To demonstrate the effectiveness of our proposed method, we compare our method with other four segmentation methods commonly used in the area of image processing, including OTSU,[51] SVM,[52] FCN,[50] and DeepLab V3+.[24] (1) The OTSU method is a typical image processing algorithm, which is widely used in image segmentation and achieves good segmentation results. (2) The SVM is a typical machine learning algorithm, which can effectively solve the task of image segmentation. In the experiment, we use color and boundary information as features, which have strong discriminability. (3) The FCN is one of the most famous DCNN for semantic segmentation. (4) The DeepLab V3+ is one of the most advanced DCNN for semantic segmentation, which has the encoder-decoder architecture and obtains the best semantics segmentation effect on ImageNet.

The comparison results are reported in Table 2. In this table, we show five performance measure results, including training loss (l_1), test loss (l_2), difference between training loss and test loss (l_d), inference time per image ($\tilde{t}$), and accuracy (acc). For easy observation, the best measure values are shown in boldface. As can be seen, the proposed method can obtain the superior performance than other methods. The typical image processing algorithm OTSU obtains the worse performance than other machine learning methods, because

Table 2 Quantitative comparison between the proposed method and other typical segmentation methods.

Method	l_1	l_2	l_d	$\tilde{t}$	Acc (%)
OTSU	—	—	—	5.0	92.28
SVM	0.1684	1.8853	1.7169	46.0	97.14
FCN-32s	5.3213	5.7243	0.3030	2.0	96.27
FCN-16s	1.5011	1.6216	0.1205	2.1	97.20
FCN-8s	0.2102	0.3233	0.1131	1.6	97.96
DeepLab V3+	0.1456	0.1532	0.0076	2.0	96.84
HPTL	**0.0141**	**0.0166**	**0.0025**	**1.1**	**98.81**

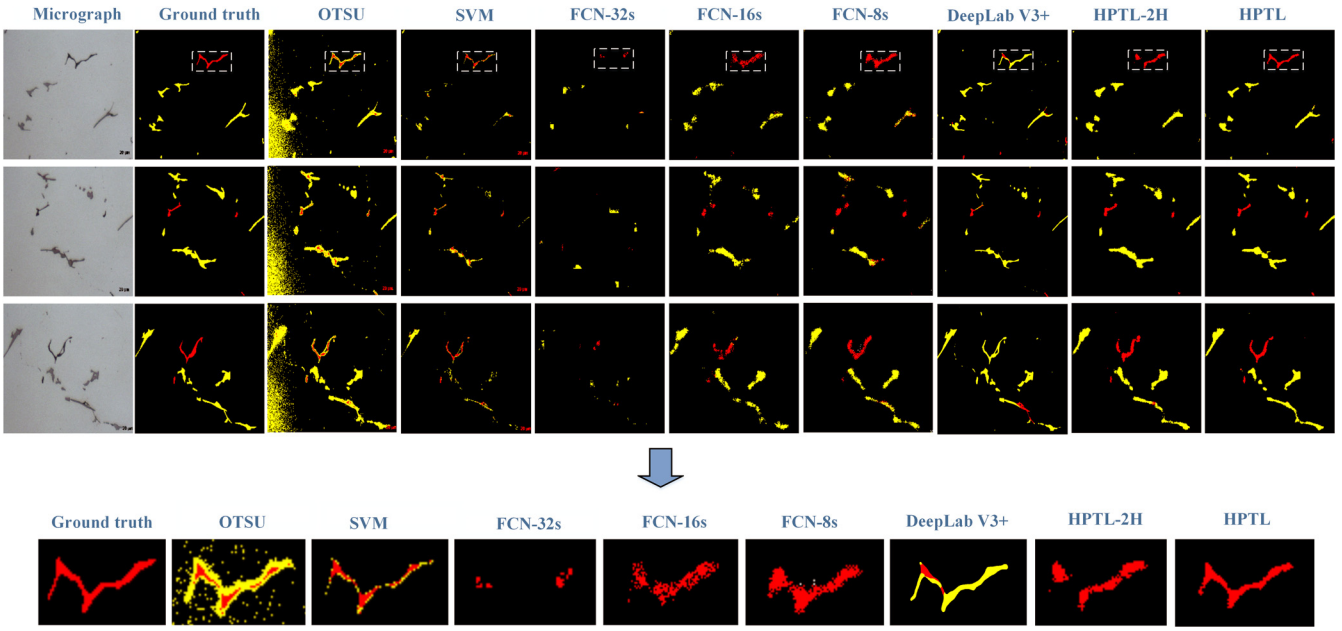

Fig. 5 Qualitative comparison between the proposed method and other typical segmentation methods.

it does not have the learning process. Since the hand-labeled data are insufficient, the results of using DeepLab V3+ with complex network to segment the aluminum alloy micrographs are not satisfactory. The SVM method can achieve satisfactory results, but we spend a lot of time on feature extraction. Thus the overall performance of the proposed method is much better than other methods for aluminum alloy micrograph dataset without sufficient hand-labeled data. Moreover, in metallographic images, some important microstructures are usually very small in area. Therefore, even when the segmentation accuracy is high, some important microstructures information will be lost, as shown in Fig. 5. In order to deal with this problem, our proposed improved methodology is very necessary.

In order to evaluate the validity of our proposed multilayer structure, we compare HPTL method with HPTL-2H method. Here the HPTL-2H method can be seen as a special form of the HPTL method, which contains two parameter transfer processes. The work flow of HPTL-2H is shown in red box in Fig. 3. The comparison results are shown in Table 3. From this table, it is obvious that the HPTL has better performance than HPTL-2H method. These experiment results verify the validity of our proposed multilayer structure.

In addition, we also evaluate the effect of regularization on HPTL. The HPTL-R represents the HPTL method with regularization item. The comparison results are listed in Table 3. From this table, we can see that the HPTL-R method has better performance than HPTL. These experiment results show that the regularization technique is helpful to improve the segmentation performance of HPTL.

Table 3 Comparison among HPTL, HPTL-2H, and HPTL-R.

Method	l_1	l_2	l_d	t	Acc (%)
HPTL-2H	0.5611	0.6721	0.1110	1.3	98.32
HPTL	0.0141	0.0166	0.0025	1.1	98.81
HPTL-R	**0.0130**	**0.0154**	**0.0024**	**0.9**	**98.88**

To further evaluate the proposed method, Fig. 5 shows the segmentation results processed by OTSU, SVM, FCN-32s, FCN-16s, FCN-8s, DeepLab V3+, HPTL-2H, and HPTL. The first column shows three original aluminum alloy micrographs, and the second column shows their corresponding ground truths. In the ground truth, black region represents aluminum, red region represents Mg_2Si, and yellow region represents Fe-containing phase. The segmentation results of different methods are shown in the different columns. For example, the segmentation results of the proposed HPTL method are shown in the last column. For convenient observation, we show the local enlarged maps of segmentation results obtained by different methods, as shown in the fourth row in Fig. 5.

As shown in the second column in Fig. 5, the OTSU method cannot achieve accurate segmentation of aluminum alloy microstructure, because the global threshold is adopted in this method. The learning-based method can solve this problem effectively. As can be seen, the SVM and DeepLabV3+ can accurately segment objects from background, but they cannot accurately distinguish different objects, such as Mg_2Si and Fe-containing phase. This problem is mainly caused by insufficient hand-labeled data. In addition, we can see that FCN-32s and FCN-16s omit a lot of microstructures in micrograph. Although FCN-8s and HPTL-2H can accurately segment objects from background, they have many mistakes in details, such as microstructural edge. Compared with other methods, our proposed method can obtain more accurate segmentation results.

5 Conclusions

In this paper, an HPTL method is proposed for the segmentation of microstructure in aluminum alloy micrograph, which can be seen as the generalization of the typical

parameter transfer learning method. In this method, we use the four different DCNNs, including VGG-19, FCN-32s, FCN-16s, and FCN-8s. The multilayer structure is designed to transfer network parameters in the order from high transferability to low transferability. The experiment results demonstrate that the proposed methodology is able to avoid the problems caused by insufficient hand-labeled data. Its overall performance outperforms other four typical segmentation methods, including state-of-the-art DeepLab V3+ method.

Acknowledgments

This work was supported by the National Key R&D Program of China (No. 2017YFB0306401) and the National Natural Science Foundation of China (No. 61773104).

References

1. E. A. Strake, Jr. and J. T. Staley, "Application of modern aluminum alloys to aircraft," *Progr. Aerosp. Sci.* **32**(2), 131–172 (1996).
2. J. H. Martin et al., "3-D printing of high-strength aluminium alloys," *Nature* **549**(7672), 365–369 (2017).
3. M. Hytch, J. Putaux, and J. Penisson, "Measurement of the displacement field of dislocations to 0.03 by electron microscopy," *Nature* **423**(6937), 270–273 (2003).
4. A. Campbell et al., "New methods for automatic quantification of microstructural features using digital image processing," *Mater. Des.* **141**, 395–406 (2018).
5. X. Zhenying et al., "Algorithm based on regional separation for automatic grain boundary extraction using improved mean shift method," *Surf. Topogr.: Metrol. Prop.* **6**(2), 25001 (2018).
6. S. Journaux et al., "Evaluating creep in metals by grain boundary extraction using directional wavelets and mathematical morphology," *J. Mater. Process. Tech.* **117**(1), 132–145 (2001).
7. Q. D. Sun et al., "Metallographical image segmentation and compression," *Appl. Mech. Mater.* **152–154**(11), 276–280 (2012).
8. J. P. Simmons et al., "Physics of MRF regularization for segmentation of materials microstructure images," in *IEEE Int. Conf. Image Process. (ICIP)*, pp. 4882–4886 (2014).
9. H.-C. Cheng, A. Cardone, and A. Varshney, "Interactive exploration of microstructural features in gigapixel microscopy images," in *IEEE Int. Conf. Image Process. (ICIP)*, pp. 335–339 (2017).
10. L. Chen et al., "Two-dimensional fuzzy clustering algorithm (2DFCM) for metallographic image segmentation based on spatial information," in *2nd Int. Conf. Inf. Sci. and Control Eng.*, pp. 519–521 (2015).
11. V. H. C. D. Albuquerque et al., "Evaluation of multilayer perceptron and self-organizing map neural network topologies applied on microstructure segmentation from metallographic images," *NDT & E Int.* **42**(7), 644–651 (2009).
12. D. S. Bulgarevich et al., "Pattern recognition with machine learning on optical microscopy images of typical metallurgical microstructures," *Sci. Rep.* **8**(1), 2078 (2018).
13. J. P. Papa et al., "Computer techniques towards the automatic characterization of graphite particles in metallographic images of industrial materials," *Expert Syst. Appl.* **40**(2), 590–597 (2013).
14. V. H. de Albuquerque et al., "Automatic evaluation of nickel alloy secondary phases from SEM images," *Microsc. Res. Tech.* **74**(1), 36–46 (2011).
15. B. L. Decost and E. A. Holm, "A computer vision approach for automated analysis and classification of microstructural image data," *Comput. Mater. Sci.* **110**, 126–133 (2015).
16. J. Gola et al., "Advanced microstructure classification by data mining methods," *Comput. Mater. Sci.* **148**, 324–335 (2018).
17. Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature* **521**(7553), 436–444 (2015).
18. H. Xing, G. Zhang, and M. Shang, "Deep learning," *Int. J. Semantic Comput.* **10**(3), 417–439 (2016).
19. Y. Lecun et al., "Gradient-based learning applied to document recognition," *Proc. IEEE* **86**(11), 2278–2324 (1998).
20. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Commun. ACM* **60**(6), 84–90 (2017).
21. F. Clement et al., "Learning hierarchical features for scene labeling," *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(8), 1915–1929 (2013).
22. K. He et al., "Deep residual learning for image recognition," in *IEEE Conf. Comput. Vision & Pattern Recognit.* (2016).
23. L. Yi et al., "Fully convolutional instance-aware semantic segmentation," in *IEEE Conf. Comput. Vision & Pattern Recognit.*, pp. 4438–4446 (2017).
24. L. Chen et al., "Encoder-decoder with atrous separable convolution for semantic image segmentation," *Lect. Notes Comput. Sci.* **11211**, 833–851 (2018).
25. A. Waibel et al., "Phoneme recognition using time-delay neural networks," *IEEE Trans. Acoust. Speech Signal Process.* **37**(3), 328–339 (1989).
26. G. E. Hinton et al., "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Process. Mag.* **29**(6), 82–97 (2012).
27. A. Graves, A. R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *IEEE Int. Conf. Acoust.*, pp. 6645–6649 (2013).
28. S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.* **9**(8), 1735–1780 (1997).
29. A. Graves, "Generating sequences with recurrent neural networks," *Comput. Sci.* (2013).
30. J. Ma et al., "Deep neural nets as a method for quantitative structure–activity relationships," *J. Chem. Inf. Model.* **55**(2), 263–274 (2015).
31. M. Helmstaedter et al., "Connectomic reconstruction of the inner plexiform layer in the mouse retina," *Nature* **500**(7461), 168–174 (2013).
32. H. Y. Xiong et al., "The human splicing code reveals new insights into the genetic determinants of disease," *Science* **347**(6218), 1254806–1254806 (2015).
33. B. Sanchezlengeling and A. Aspuruguzik, "Inverse molecular design using machine learning: generative models for matter engineering," *Science* **361**(6400), 360–365 (2018).
34. A. Chowdhury et al., "Image driven machine learning methods for microstructure recognition," *Comput. Mater. Sci.* **123**, 176–187 (2016).
35. N. Lubbers, T. Lookman, and K. Barros, "Inferring low-dimensional microstructure representations using convolutional neural networks," *Phys. Rev. E* **96**(5–1), 052111 (2017).
36. S. M. Azimi et al., "Advanced steel microstructure classification by deep learning methods," *Sci. Rep.* **8**(1), 2128 (2018).
37. B. Ma et al., "Deep learning-based image segmentation for al-la alloy microscopic images," *Symmetry* **10**(4), 107 (2018).
38. S. Zhang et al., "Aluminum alloy microstructural segmentation method based on simple noniterative clustering and adaptive density-based spatial clustering of applications with noise," *J. Electron. Imaging* **28**(3), 33035 (2019).
39. J. West, D. Ventura, and S. Warnick, "Spring research presentation: a theoretical foundation for inductive transfer," Brigham Young University, College of Physical and Mathematical Sciences 1 (2007).
40. S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.* **22**(10), 1345–1359 (2010).
41. C. Tan et al., "A survey on deep transfer learning," in *Int. Conf. Artif. Neural Networks*, pp. 270–279 (2018).
42. J. Yosinski et al., "How transferable are features in deep neural networks?" in *Proc. 27th Int. Conf. Neural Inf. Process. Syst.*, Vol. 27, pp. 3320–3328 (2014).
43. T. Liu, Q. Yang, and D. Tao, "Understanding how feature structure transfers in transfer learning," in *Twenty-Sixth Int. Joint Conf. Artif. Intell.*, pp. 2365–2371 (2017).
44. A. Zamir et al., "Taskonomy: disentangling task transfer learning," in *Comput. Vision and Pattern Recognit.* (2018).
45. E. Tzeng et al., "Deep domain confusion: maximizing for domain invariance," *Comput. Sci.* (2014).
46. M. Long and J. Wang, "Learning transferable features with deep adaptation networks," in *Int. Conf. Mach. Learn.*, pp. 97–105 (2015).
47. J. Devlin et al., "BERT: pre-training of deep bidirectional transformers for language understanding," arXiv:1810.04805 (2018).
48. C. B. Do and A. Y. Ng, "Transfer learning for text classification," in *NIPS*, pp. 299–306 (2005).
49. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *Comput. Sci.* (2014).
50. J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, pp. 3431–3440 (2015).
51. N. Otsu, "A threshold selection method from gray level histograms," *IEEE Trans. Syst. Man Cybern.* **9**(1), 62–66 (1979).
52. J. C. Platt, "Fast training of support vector machines using sequential minimal optimization," in *Advances in Kernel Methods*, pp. 185–208, MIT Press, Cambridge, Massachusetts (1999).

Dali Chen received his PhD in pattern recognition and intelligent systems from Northeastern University, China, in 2008. He is an associate professor in the College of Information Science and Engineering at Northeastern University, China. His research lies at the intersection of machine learning and image processing. His current research interest is to develop deep learning algorithms for medical image processing and industrial intelligent systems.

Pengyuan Zhang received his BS degree in building electrical and intelligence from Shenyang Architectural University, Liaoning,

China, in 2017. He is a postgraduate student in the College of Information Science and Engineering, Northeastern University. His current research interests include image processing and deep learning methods.

Shixin Liu received his BS degree in mechanical engineering from Southwest Jiaotong University, Sichuan, China, in 1990, and his MS and PhD degrees in systems engineering from Northeastern University, Shenyang, China, in 1993 and 2000, respectively. He is a professor in the College of Information Science and Engineering at Northeastern University, China. His research interests are in intelligent optimization algorithms, planning and scheduling, machine learning, and computer vision.

Yangquan Chen received his MS degree in automatic control from Beijing Institute of Technology, China, in 1989 and his PhD in electrical engineering from Nanyang Technological University, Singapore, in 1998. He is an associate professor in the School of Engineering, University of California, Merced, USA. His current areas of research interest include smart mechatronics and controls (intelligent, optimal, robust, nonlinear, and adaptive).

Wei Zhao received his PhD degree in material processing engineering from Huazhong University of Science and Technology, China, in 2013. He is an assistant dean of the research institute in Shandong Nanshan Aluminum Industry Co. Ltd., China. His research interests are in the fields of materials science and technology.