arXiv:1905.06656v1 [cs.CV] 16 May 2019

One-Shot Texture Retrieval with Global Context Metric

Kai Zhu, Wei Zhai, Zheng-Jun Zha, Yang Cao[*]
University of Science and Technology of China
{zkzy, wzhai056}@mail.ustc.edu.cn, {zhazj, forrest}@ustc.edu.cn

Abstract

In this paper, we tackle one-shot texture retrieval: given an example of a new reference texture, detect and segment all the pixels of the same texture category within an arbitrary image. To address this problem, we present an OS-TR network to encode both reference and query image, leading to achieve texture segmentation towards the reference category. Unlike the existing texture encoding methods that integrate CNN with orderless pooling, we propose a directionality-aware module to capture the texture variations at each direction, resulting in spatially invariant representation. To segment new categories given only few examples, we incorporate a self-gating mechanism into relation network to exploit global context information for adjusting per-channel modulation weights of local relation features. Extensive experiments on benchmark texture datasets and real scenarios demonstrate the above-par segmentation performance and robust generalization across domains of our proposed method.

1 Introduction

As texture refers to the fundamental microstructures of natural image, humans have a strong visual perception of texture, which can not only obtains descriptions of new texture from a small number of training samples (few-shot learning) [Sung *et al.*, 2018], but also marks such texture regions in the other cluttered scenes (texture segmentation) [Cimpoi *et al.*, 2015]. This suggests that texture features provide a powerful visual prior for comprehensive scene understanding [Krishna *et al.*, 2017].

To learn such texture prior, we present the problem of one-shot texture retrieval: given an example of a new reference texture, detect and segment all the pixels of the same texture category within an arbitrary image (see Figure 1). This task is different from one-shot segmentation of general objects [Shaban *et al.*, 2017], as the learned texture representation should be invariant to spatial layout but preserve the rough semantic concepts. Therefore, an adaptable and robust texture encoding model should be presented to finely discriminate the orderless texture details. In addition, for texture segmentation, global context is also an important cue since the scene surfaces are usually not completely orderless. A similarity metric should be introduced to balance local spatial details and global scene context for pixel-wise segmentation.

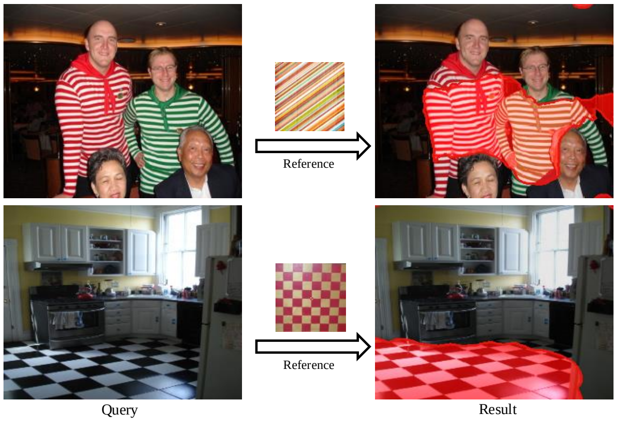

Figure 1: Examples of our task. Given a reference texture of new category, we can segment all pixels of the same texture category within a query image.

In this paper, we present an One-Shot Texture Retrieval (OS-TR) network to learn the texture representation, and model the similarities between reference and query image, leading to achieve one-shot texture segmentation towards the reference category. Specifically, a Siamese network [Koch *et al.*, 2015]is first presented to embed the reference and query image into an encoded representation space by feature learning and parametric prior encoding. Unlike the existing texture encoding methods [Xue *et al.*, 2018] that integrate CNN with orderless pooling, a directionality-aware module is proposed to perceive the local texture variations at each direction, resulting in spatially invariant representation. Then, we propose to incorporate global context into the relation network [Sung *et al.*, 2018] by aggregating feature maps across their spatial dimensions. Different from previous approaches that only consider the similarity of local features or semantic concepts, our method exploits global context information to adjust local relation features by per-channel modulation weights. The key idea is to use a self-gating mecha-

[*]Corresponding author

nism for generating global distribution of local relation features with an aggregation unit. The evaluation on synthetic dataset demonstrates the superiority of our proposed model against the state-of-the-art methods [Shaban *et al.*, 2017; Ustyuzhaninov *et al.*, 2018]. Furthermore, the result on nature scenes is promising.

Our main contribution are as follows:

1. We introduce a novel one-shot texture segmentation network with global context metric, achieving texture detection and segmentation with a single example of reference texture.

2. We propose a directionality-aware module to perceive the local texture variations at each direction, resulting in spatially invariant representation.

3. We present a global context metric for performing one-shot texture segmentation, which extends relation network with a self-gating mechanism to adjust local relation features.

2 Related Work

Our work focuses on one-shot learning, texture modeling and one-shot segmentation task, so in this section we mainly review the research status of these areas.

One-shot learning: In the computer vision community, one-shot learning has recently received a lot of attention and substantial progress has been made based on metric learning using Siamese neural networks [Koch *et al.*, 2015; Snell *et al.*, 2017]. In addition, there are some work that build upon meta learning [Finn *et al.*, 2017; Ren *et al.*, 2018], information retrieval techniques [Triantafillou *et al.*, 2017] and generation models [Lake *et al.*, 2015] to achieve one-shot learning.

Texture representation: Texture representation is an important research area in computer vision for potential applications in classification, segmentation and synthesis. The research of texture representation are mainly divided two classes: traditional method [Kumar *et al.*, 2011] and CNN-based method [Cimpoi *et al.*, 2015]. Different from object recognition where spatial order is critical for feature representation, texture recognition usually uses an orderless component to provide invariance to spatial layout.

One-shot segmentation: While the work on one-shot learning is quite extensive, the research on one-shot segmentation [Dong and Xing, 2018; Wu *et al.*, 2018] have been established only recently, including one-shot image segmentation [Shaban *et al.*, 2017] and one-shot video segmentation [Caelles *et al.*, 2018]. The most closely related to our work is [Ustyuzhaninov *et al.*, 2018], whose task is to segment an input image containing multiple textures by given a patch of a reference texture. Different from their setup in that the reference patch is interactively selected from the input image, our work targets on a more complex problem of one-shot texture retrieval: given an example of a new reference texture, detect and segment all the pixels of the same texture category within an arbitrary image.

3 One-shot Texture Segmentation

3.1 Probelm Setup

We define a triple $Tr_i^j = (Q_i, R_i^j, T_i^j)$ and a relation function $F : A_i^j = F(Q_i, R_i^j; \theta)$, where Q_i is the i^{th} query image with multiple classes of textures, T_i^j is the pixel-wise labels corresponding to the j^{th} class texture in Q_i, R_i^j is a reference texture image of the same class j, A_i^j is the actual segmentation result, and θ is all parameters to be optimized in function F. Our task is to randomly sample triples from the dataset, train and optimize θ, thus minimizing the loss function L:

$$\theta_* = arg \min_{\theta} L(A_i^j, T_i^j). \tag{1}$$

We expect that the relationship function F can segment the same class texture region in another target image each time it sees a reference texture picture of new class. This is the embodiment of the meaning of one-shot segmentation. It should be mentioned that the texture classes sampled by the test set are not present in the training set, that is, $U_{train} \bigcap U_{test} = \emptyset$. The relation function F in this problem is implemented by the model detailed in Section 3.2.

3.2 Model Architecture

In this section we will detail the overall framework of the proposed OS-TR network. As shown in Figure 2, the network is based on a classic encoder-decoder network [Ronneberger *et al.*, 2015] to complete the segmentation task. It consists of the texture encoder for captioning the characteristics of the texture and global context metric that better determines the matching pixels. These two modules will be explained further in Section 3.3 and 3.4.

The network uses the texture encoders f_t to transform the input of two branches into the variable embeddings respectively. The first branch takes the reference texture R_i^j from the triple as input. And the second branch takes the synthetic texture image as input during training, which may be from a wide range of sources in the real scenarios. We use the texture encoder to perceive the local texture features at each dirction, and the corresponding feature map pairs M_1, M_2 obtained can be expressed as follows:

$$M_1 = f_t(R_i^j; \theta_t), \tag{2}$$

$$M_2 = f_t(Q_i^j; \theta_t). \tag{3}$$

Here θ_t is the learnable parameter of our texture encoder.

Different from the existing work, one-shot relation network [Sung *et al.*, 2018] proposes a deep nonlinear metric. However, this nonlinaer comparator is limited to local ralation features. Instead we learn a global context metric g_c to compare the pixel information of query images with the reference texture by taking global context in consideration. It provides important features for subsequent pixel-level segmentation. In our network,

$$S = g_c(M_1, M_2; \theta_c), \tag{4}$$

where S refers to the element-wise relation score, and θ_c is the parameter to be optimized in g_c.

To generate the same size as the original image, we combine the score S with the information of encoder f_t in multiple stages. We make full use of extracted features at different scales in the encoding process to form a refined decoding layer g_d. The final actual segmentation result A of the original image is obtained through a sigmoid function:

$$A = Sigmoid(g_d(S, f_{t_2}; \theta_d)). \tag{5}$$

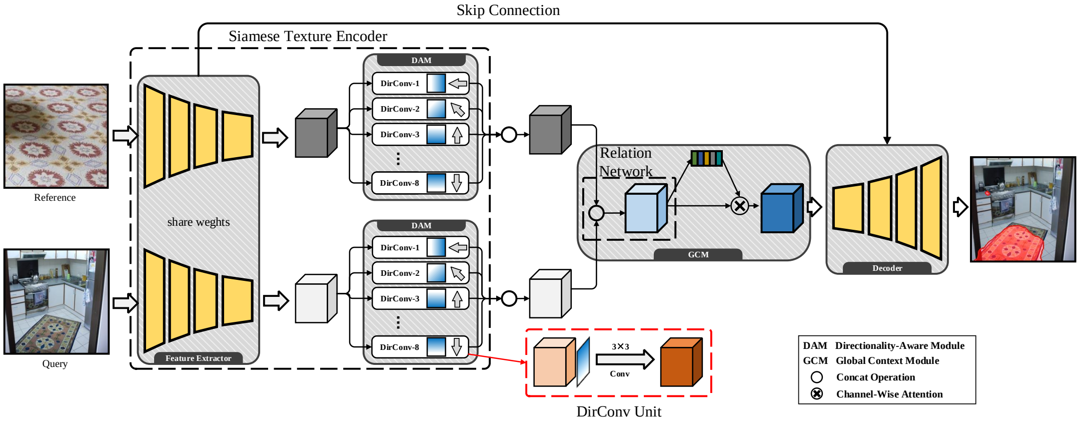

Figure 2: Overview of the OS-TR network. The texture features of the query image and reference are first extracted by the proposal Siamese texture encoder. Through a global context metric, the relation score is obtained by means of global information and then combined with the encoding features to get the final segmentation results.

Similarly, θ_d stands for the parameters of decoding part.

The number of positive and negative samples in the training set is unbalanced (i.e., the foreground and background of query images). To ensure their equal contribution to optimization function, we use weighted Binary Cross Entory Loss function, that is

$$\begin{aligned} L = -\alpha \sum_{p \in T_+} logPr(y_p = 1 \mid Q_i, R_i^j; \theta) - \\ (1-\alpha) \sum_{p \in T_-} logPr(y_p = 0 \mid Q_i, R_i^j; \theta), \end{aligned} \tag{6}$$

where y_p stands for the ground truth of corresponding pixel p, $\alpha = |T_-|/(|T_-| + |T_+|)$. T_+ and T_- denotes positive and negtive sample sets in training images, respectively. We can see θ in the loss function is the collection of θ_t, θ_c and θ_d described earlier. The purpose of reducing loss function is to optimize the parameters of corresponding modules.

3.3 Texture Encoder

To perceive the local texture variations at each direction, we propose the texture encoder consisting of feature extraction and directionality-aware module. As shown in Figure 2, ResNet (ResNet in this paper represents ResNet-50 from [He *et al.*, 2016] that removes the last fully connected layer) is used to extract preliminary features P_1, P_2 from the query image and reference texture. Then the DirConv unit g_d is presented to capture eight directional texture features G_i under the guidance of corresponding directional feature map D_i. We only detail the first branch as the parameters and structures of two branches are identical. Specifically,

$$G_i = g_d(cat(P_1, D_i))(i = 1, 2 \cdots 8), \tag{7}$$

where g_d denotes a convolution block which contains a 3×3 convolution, a batch normalization and a ReLU activation layer, cat refers to the concatenation function in the channel dimension. In this paper, eight directions refer to top, bottom, left, right, top left, bottom left, top right, and bottom right. Each directional map D_m is a generated trend graph that decreases from 1 to 0 along a certain direction. Finally, the output features of different branches are concatenated to form the whole spatial invariant feature M_1, that is,

$$M_1 = g_h(cat(G_1, G_2 \cdots G_8)), \tag{8}$$

where g_h represents a set of standard convolution block. Since the proposed DirConv unit is sensitive to the local variation of image along each direction, it can provide the network with good adaptability to spatial distortion and scale changes.

Layer	Input	Output	Type
Feature Extractor			
ResNet-50: $conv1 - conv5$			
Directionality-Aware Module			
DirConv unit	$8 \times 8 \times$ $(2048 + \mathbf{1})$	8×8 $\times 256$	3×3 Conv
Join	$8 \times$ $[8 \times 8 \times 256]$	8×8 $\times 2048$	Cat
Global Context Metric			
Join	$2 \times$ $[8 \times 8 \times 2048]$	8×8 $\times 1024$	Cat+ 3×3 Conv
Aggregation	$8 \times 8 \times 1024$	8×8 $\times 1024$	Max pooling + 1×1 Conv
Weighting	$8 \times 8 \times 1024$	8×8 $\times 1024$	Channel-wise Multiplication
Decoder			
Upsample ($\times 4$)	$8 \times 8 \times 1024$	256×256 $\times 1$	Bilinear Interpolation + 1×1 Conv + Sigmoid

Table 1: Details of our network. '×4' represents four upsampling operations. The bold **1** in the DirConv unit represents a directional map.

3.4 Global Context Metric

To achieve pixel-wise segmentation, we incorporate global context into local relation feature to adjust per-channel modulation weights. Firstly, we use a non-linear function g_m to

U_{test_i}	classes
i=0	perforated, pitted, pleated, polka-dotted, porous
i=1	stained, stratified, striped, studded, swirly
i=2	veined, waffled, woven, wrinkled, zigzagged

Table 2: Specific class names of three test sets defined in section 4.1.

Method	i=0	i=1	i=2	mean
Baseline	56.4	47.2	44.9	49.5
+Tex	59.3	50.3	**46.4**	52.0
+Glo	59.4	51.5	45.4	52.1
+Tex& Glo	**60.7**	**52.8**	44.8	**52.8**

Table 3: Results of ablation study. 'Tex' and 'Glo' represent texture encoder and global context metric respectively. The middle three columns represent the mean IoU metric(%) for the three test sets, and the mean represents the average of results for the three test sets.

capture local relation features L:

$$L = g_m(cat(M_1, M_2)), L \in \mathbb{R}^{H\times W\times C} \tag{9}$$

where g_m represents two sets of standard convolution blocks, $H \times W$ and C refers to spatial and channel dimensions of the feature map, respectively. It can be seen that the two-layer convolution module represents more metric possibilities, which is not limited by cosine, Euclidean distance [Vinyals *et al.*, 2016; Snell *et al.*, 2017], etc. However, it only considers local feature similarity which has its limitation. Instead we take global context [Qiao *et al.*, 2019] into consideration by aggregating feature maps across their spatial dimensions, which is similar to SENet [Hu *et al.*, 2018].

Let $L = [l_1, l_2 \cdots l_C]\,(l_i \in \mathbb{R}^{H\times W})$ denote the local relation features of different channels. In this paper, we simply obtain the global information of each channel $\beta \in \mathbb{R}^C$ through maximum pooling g_{max}. The aggregation unit can be represented as follows:

$$\beta_i = g_{max}(l_i), i = 1, 2 \cdots C. \tag{10}$$

Next we use the collected global context to balance the relation features. Here we use two simple 1×1 convolutional blocks g_s to achieve per-channel modulation weights. Finally, the obtained weights are normalized to 0-1 and used as multiplication coefficient of corresponding channel of original feature L. The re-layout of local relation features can be formulated as follows:

$$S = \gamma \cdot L = sigmoid(g_s(\beta)) \cdot L, \tag{11}$$

where S is the same as the one in Equation 4 and $\cdot$ represents the channel-wise multiplication. It can be seen from the visualization of experimental part that global context metric realizes the fine adjustment of different texture matching pairs. It is beneficial for the optimization of the final segmentation result.

4 Experiments

To validate the superiority of our model in one-shot texture segmentation task, we designed a series of experiments based on Describable Textures Dataset(DTD) [Cimpoi *et al.*, 2014] dataset. In this section, we first introduce the preprocessing process of the dataset, and then perform ablation experiments on main modules in the model. While demonstrating its superiority from objective indicators, we also give some good visualization results. Finally, we compare our model to the most advanced model One-Shot Learning for Semantic Segmentation (OSLSM) [Shaban *et al.*, 2017] and One-shot Texture Segmentation (OSTC) [Ustyuzhaninov *et al.*, 2018] in the current one-shot segmentation field. We also introduce expanded experimental contents in supplementary materials[1].

4.1 Dataset and Setting

To solve the proposed one-shot texture segmentation task, we redivide DTD dataset into training and test sets. We divide the last 15 classes of DTD dataset into three test subsets on average, and the remaining classes are used as the training set. The specific class name is shown in Table 2. During the training, we randomly sample 2-5 texture images from the training set to synthesize query images (these textures may come from the same class), and generate labels for one of the texture regions. The technique of texture synthesis comes from [Ustyuzhaninov *et al.*, 2018]. Finally, we randomly sample the reference texture image from the training set with the same kind of labels to form the triple defined in Section 4.1. In the test phase, we synthesize 240 query images (a total of 960 images) from the three subtest sets in the same way. We send them to the trained model to calculate evaluation indicators and then average them to ensure that different texture classes in the test set are relatively fair. The regular evaluation indicator IoU is used in our task.

Our model uses the SGD optimizer during the training process. The initial learning rate is set to 0.001 and the attenuation rate is set to 0.0005. The model stops training after 1000 epochs, where each epoch synthesizes 240 query images. All images are resized to 256×256 size and the batch size is set to 16.

4.2 Ablation Study

We conduct several ablation experiments to prove the effectiveness of directionality-aware module and global context metric of our model in Table 3. We first train a baseline without the two modules, and then add them respectively to compare the results. First of all, we can see that the global context metric improves the model by 2.6% mean IoU. This is due to its use of global information to help the model adjust the overall feature distribution. In order to explain this more vividly, we make a visualization of per-channel modulation weight distribution learned by global context metric in Section 4.3. Then, the result has a 2.5% mean IoU boost with the dierctionality-aware module. This module takes into account the characteristics of texture spatial distribution, which is very helpful for texture encoding. The specific analytical experiment is also shown in Section 4.3. Finally, we add both the two modules to form our OS-TR network, which has a 3.3% improvement over the baseline.

[1] https://github.com/zhukaii/ijcai2019

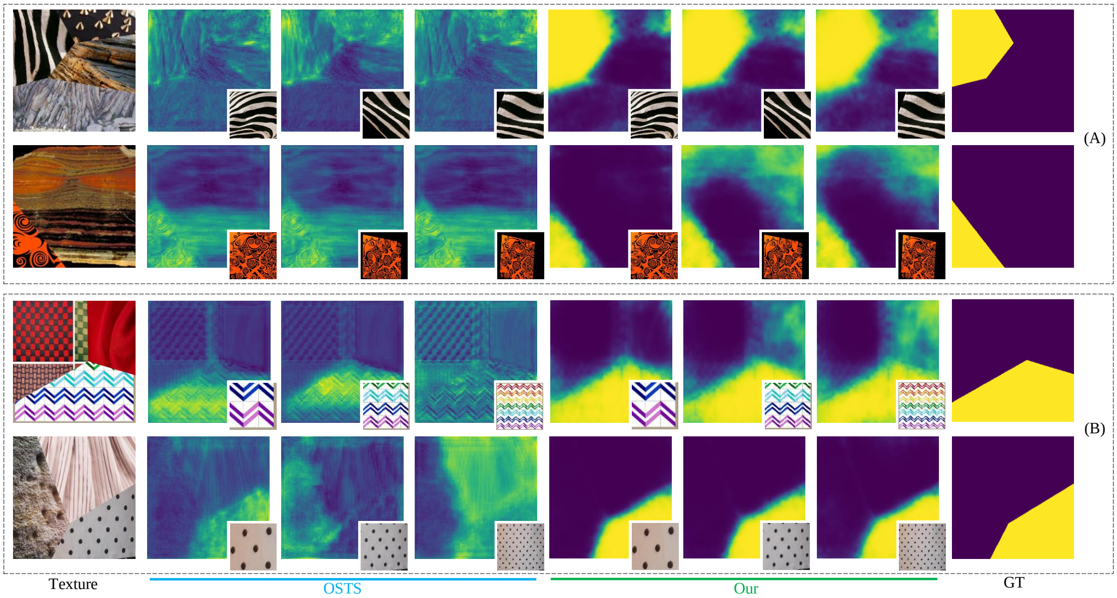

Figure 3: Verification results of spatial invariance. Part (A) and (B) represent the results of affine transformation and scale change respectively. From left to right, they represent query image, results of OSTC compared to ours and groud truth respectively. Reference textures are shown in the lower-right corner of each image of results.

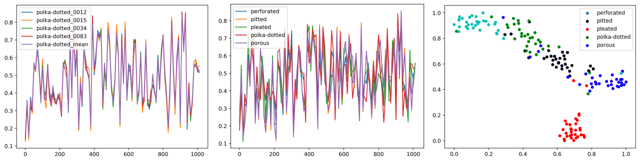

Figure 4: Illustration of the role of global context. The abscissa represents the channel and the ordinate represents the respective weight between 0 and 1 in the two figures on the left. The figure on the right is visualized by t-SNE dimensionality reduction. See detailed explanation in section 4.3.

Method	$i = 0$	$i = 1$	$i = 2$	mean
OSTC	28.0	34.0	33.6	31.9
OSLSM	57.5	47.2	42.6	49.1
Ours	**60.7**	**52.8**	**44.8**	**52.8**

Table 4: Mean IoU(%) of our model and other state-of-the-art methods.

4.3 Evaluation

Spatial Invariance: To evaluate the performance of our model on the spatial invariant texture representation, we conduct the following experiments. First, in order to demonstrate the adaptability of our model to spatial distortion, we show its segmentation effect by affining the reference texture image. As shown in Figure 3 (A), we take several query images as examples and randomly determine the parameters of affine transformation to compare the segmentation results. It can be seen the results achieve the segmentation effect similar to the original reference texture. As shown in Figure 3 (B), we select three scale reference textures 256 × 256, 128 × 128 and 64 × 64 to evaluate the sensitivity of the model to scale change. The segmentation effect is still excellent. It is the proposed DirConv unit that is sensitive to the local variation of image along each direction, so that the spatial arrangement of texture can be accurately extracted.

Effectiveness of Global Context Metric: To demonstrate the function of the module, we visualize per-channel modulation weights for different reference images. We cite five categories in our test set as examples. As shown in two figures on the left in Figure 4, all the polylines of the same class in the left picture are almost identical and the right of five classes is different. To further explain this, we reduce the weights of these five classes and visualize them through t-SNE [Maaten and Hinton, 2008]. It can be seen these five classes are clearly distinguished in the relation feature space. We think our module has learned an adaptive weight adjustment method among channels, where the intra-class is similar and inter-class appears inconsistent. As analyzed in Section 3.4, this module combines global information to further separate different matching pairs in feature space.

Comparision with state-of-the-art: To better assess the overall performance of our reference network, we compare it to OSLSM and OSTC models. Because the tasks and model backbones are different, we set all the backbones to ResNet for fair comparision, and reproduce them with pytorch according to the two articles. All models are trained and tested in the same steps to achieve the purpose of adapting to our task. OSLSM is the first solution to one-shot semantic segmentation task, which is similar to our task. So after modifying backbone, we train their models directly in our tasks and don't need to modify any dataset settings. To compare with

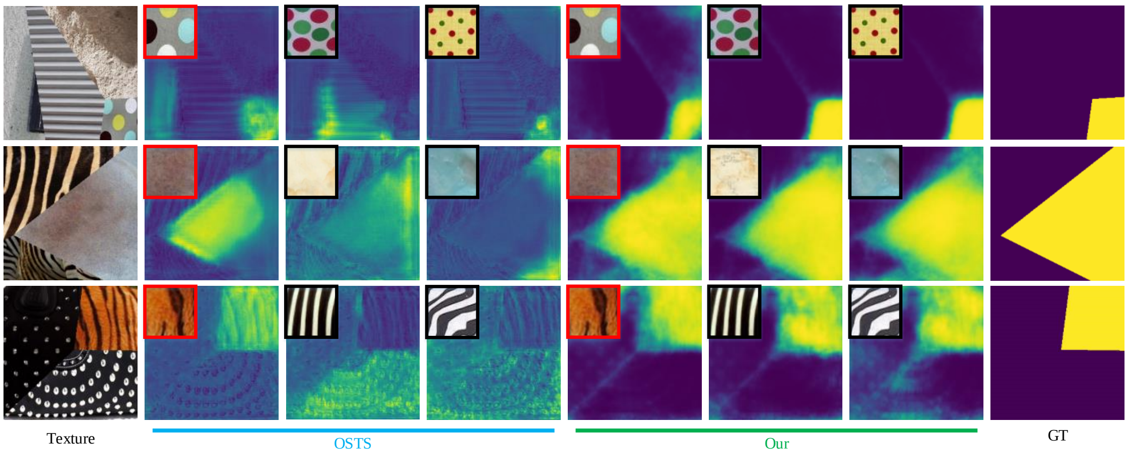

Figure 5: Comparison of our segmentation results with OSTC. Three rows represent three groups of results. From left to right, they represent query image, results of OSTC compared to ours and groud truth respectively. Reference textures are show in the upper-left corner of each image of results. The reference images with red borders are from the original query image, while reference images with black borders are from the same class textures in DTD dataset.

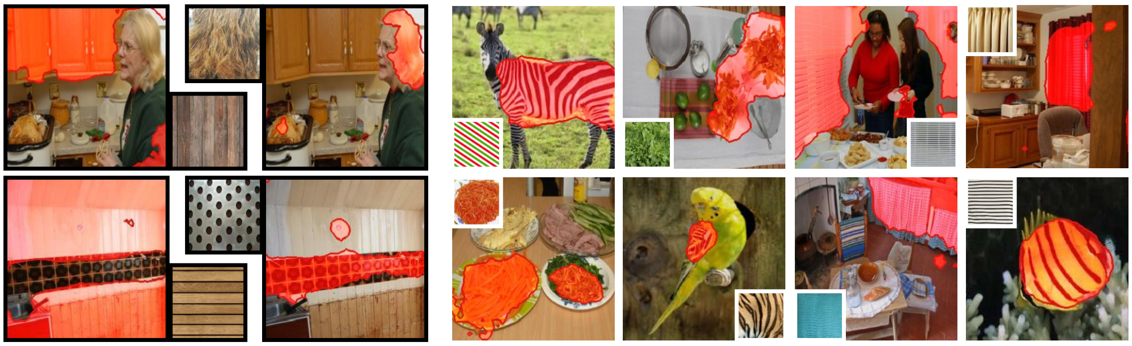

Figure 6: Some qualitative results of our model in practical application. The reference images of each picture are placed in small graphics, and the corresponding segmentation results are marked in red.

OSTC, we change the reference image to a 64×64 patch for training in our task.

As can be seen from Table 4, our model has more than three points mean IoU boost over the best performing OSLSM. Since OSTS is essentially designed for the interactive texture segmentation task, it does not work well for one-shot segmentation task. So we reproduce the model they use to solve their interactive texture segmentation task, which follows the settings of their papers. The reference texture of our model is the same class as a texture area in query image, and of course it may be the same texture image. Conversely, OSTS is not necessarily able to segment textures of the same class. As shown in Figure 5, the reference texture image in our model can achieve better segmentation effect whether it is the original image of a texture region of the query image or a different image of the same class. It benefits from the texture characteristics and the distinction of different texture features acquired by our texture module and global context metric respectively.

Evaluation on real scenarios: Our model not only performs well in synthetic texture images, but also has scalability in practical applications. To prove this, we replace the query iamges with ones taken from real scenarios. We download high-quality indoor scene pictures with rich texture information from OpenSurfaces dataset [Bell *et al.*, 2013], and select animal and plant pictures with describable texture features from the Internet. For reference images, we randomly select images with similar texture information from DTD dataset and Internet according to query image content. For example, a striped image from DTD is randomly selected as a reference for zebras. In testing the above natural images, we did not pre-train or adjust the model. The parameters are still fixed in the state trained in synthetic texture images. As shown in Figure 6, We still get good segmentation results. It can be seen that our model can really learn new texture properties quickly, which is helpful for comprehensive scene understanding.

5 Conclusion

In this paper, a novel one-shot texture segmentation network OS-TR is proposed. By using a directionality-aware module to perceive texture variations at each direction and adjusting local features with global context information, OS-TR extracts the pixel information related to given texture in query images. In addition, our model can be well generalized from synthetic images to real scenarios without any adjustment. Experimental results show that our model is superior in both performance and adaptability with respect to existing methods.

References

[Bell *et al.*, 2013] Sean Bell, Paul Upchurch, Noah Snavely, and Kavita Bala. Opensurfaces: A richly annotated catalog of surface appearance. *ACM Transactions on Graphics (TOG)*, 32(4):111, 2013.

[Caelles *et al.*, 2018] Sergi Caelles, Alberto Montes, Kevis-Kokitsi Maninis, Yuhua Chen, Luc Van Gool, Federico Perazzi, and Jordi Pont-Tuset. The 2018 davis challenge on video object segmentation. *arXiv preprint arXiv:1803.00557*, 1(2), 2018.

[Cimpoi *et al.*, 2014] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *IEEE Conference on Computer Vision & Pattern Recognition*, 2014.

[Cimpoi *et al.*, 2015] Mircea Cimpoi, Subhransu Maji, and Andrea Vedaldi. Deep filter banks for texture recognition and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3828–3836, 2015.

[Dong and Xing, 2018] Nanqing Dong and Eric P Xing. Few-shot semantic segmentation with prototype learning. In *BMVC*, volume 1, page 6, 2018.

[Finn *et al.*, 2017] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1126–1135. JMLR. org, 2017.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[Hu *et al.*, 2018] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[Koch *et al.*, 2015] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML Deep Learning Workshop*, volume 2, 2015.

[Krishna *et al.*, 2017] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017.

[Kumar *et al.*, 2011] Neeraj Kumar, Alexander Berg, Peter N Belhumeur, and Shree Nayar. Describable visual attributes for face verification and image search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(10):1962–1977, 2011.

[Lake *et al.*, 2015] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

[Maaten and Hinton, 2008] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[Qiao *et al.*, 2019] Tingting Qiao, Jing Zhang, Duanqing Xu, and Dacheng Tao. Mirrorgan: Learning text-to-image generation by redescription. *arXiv preprint arXiv:1903.05854*, 2019.

[Ren *et al.*, 2018] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B Tenenbaum, Hugo Larochelle, and Richard S Zemel. Meta-learning for semi-supervised few-shot classification. *arXiv preprint arXiv:1803.00676*, 2018.

[Ronneberger *et al.*, 2015] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[Shaban *et al.*, 2017] Amirreza Shaban, Shray Bansal, Zhen Liu, Irfan Essa, and Byron Boots. One-shot learning for semantic segmentation. *arXiv preprint arXiv:1709.03410*, 2017.

[Snell *et al.*, 2017] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, pages 4077–4087, 2017.

[Sung *et al.*, 2018] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1199–1208, 2018.

[Triantafillou *et al.*, 2017] Eleni Triantafillou, Richard Zemel, and Raquel Urtasun. Few-shot learning through an information retrieval lens. In *Advances in Neural Information Processing Systems*, pages 2255–2265, 2017.

[Ustyuzhaninov *et al.*, 2018] Ivan Ustyuzhaninov, Claudio Michaelis, Wieland Brendel, and Matthias Bethge. One-shot texture segmentation. *arXiv preprint arXiv:1807.02654*, 2018.

[Vinyals *et al.*, 2016] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Advances in neural information processing systems*, pages 3630–3638, 2016.

[Wu *et al.*, 2018] Zheng Wu, Ruiheng Chang, Jiaxu Ma, Cewu Lu, and Chi-Keung Tang. Annotation-free and one-shot learning for instance segmentation of homogeneous object clusters. *arXiv preprint arXiv:1802.00383*, 2018.

[Xue *et al.*, 2018] Jia Xue, Hang Zhang, and Kristin Dana. Deep texture manifold for ground terrain recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 558–567, 2018.