A Implementation Details of SNNs

Spiking Neuron Model

For all SNNs, we use the Integrate-and-Fire (IF) model as the spiking neuron model, which acts as the activation layer in neural networks. As mentioned in (Fang et al. 2021b,a), V_t, X_t and S_t denote the state (membrane voltage), input (current) and output (spike) of the spiking neuron model respectively at time-step t, and the dynamics of the IF model can be described as follows:

$$H_t = V_{t-1} + X_t, \tag{1}$$

$$S_t = \Theta(H_t - V_{thresh}), \tag{2}$$

$$V_t = H_t(1 - S_t) + V_{reset}S_t. \tag{3}$$

While V_t is the membrane voltage after the trigger of a spike, H_t is also the membrane voltage, but after charging and before a spike firing. $\Theta(x)$ is the unit step function, so S_t equals 1 when H_t is greater than or equal to the threshold voltage V_{thresh} and 0 otherwise. Meanwhile, when a spike fires, V_t is reset to V_{reset}. Here, we set $V_{thresh} = 1$ and $V_{reset} = 0$.

In addition, because $\Theta(x)$ is non-differentiable at 0, the surrogate gradient method (Neftci, Mostafa, and Zenke 2019) is applied to approximate the derivative function during back-propagation. Here, we use the inverse tangent function as the surrogate gradient function

$$\sigma(x) = \frac{1}{\pi}\arctan(\pi x) + \frac{1}{2}, \tag{4}$$

and the derivative function is

$$\sigma'(x) = \frac{1}{1 + (\pi x)^2}. \tag{5}$$

Architectures of SNNs

In our experiments on SNNs, we not only use SEW ResNet proposed by (Fang et al. 2021a), but also build several new SNNs. On the one hand, we improve the spike-element-wise block in SEW ResNet with new architectures referring to studies on ResNet (He et al. 2016; Zagoruyko and Komodakis 2016; Xie et al. 2017), as shown in Table 1. On the other hand, as the multi-branch structures in CNNs increase neural representation similarity to mouse visual cortex, we use depthwise separable convolutions and follow the overall architecture of MobileNetV2 (Howard et al. 2017; Sandler et al. 2018) to build the SpikingMobileNet, the basic block of which is shown in Figure 1.

Our implementation is based on SpikingJelly (Fang et al. 2020), an open-source framework of deep SNN.

Hyper-Parameters in SNNs' Training

We use the ImageNet dataset to pre-train the new SNNs. Following the settings for training SEW ResNet (Fang et al. 2021a), we train the models for 320 epochs on 8 GPUs (NVIDIA V100), using SGD with a mini-batch size of 32. The momentum is 0.9 and the weight decay is 0. The initial learning rate is 0.1 and we decay it with a cosine annealing, where the maximum number of iterations is the same as the number of epochs. For all SNNs, we set the simulation duration $T = 4$.

B Overall model rankings

The results of model rankings are shown in Figure 2, 3 and 4. We also apply the Spearman's rank correlation to the overall model rankings of different metrics, which is shown in Figure 5.

C Score Comparisons among Model Groups

We conduct comparisons of similarity scores among CNNs, SNNs, and vision transformers. The results are shown in Figure 6.

D Overall CNN rankings

The results of CNN rankings are shown in Figure 7, 8 and 9.

E Correlations between the Model Sizes and the Similarity Scores

The results of linear regression to model sizes and the similarity scores are shown in Figure 10, 11 and 12.

F The ImageNet Accuracy and the Similarity Scores

The results are shown in Figure 13.

References

Fang, W.; Chen, Y.; Ding, J.; Chen, D.; Yu, Z.; Zhou, H.; Masquelier, T.; Tian, Y.; and other contributors. 2020. SpikingJelly. https://github.com/fangwei123456/spikingjelly. Accessed: 2022-07-06.

Fang, W.; Yu, Z.; Chen, Y.; Huang, T.; Masquelier, T.; and Tian, Y. 2021a. Deep residual learning in spiking neural networks. In *Advances in Neural Information Processing Systems 34*, 21056–21069.

Fang, W.; Yu, Z.; Chen, Y.; Masquelier, T.; Huang, T.; and Tian, Y. 2021b. Incorporating learnable membrane time constant to enhance learning of spiking neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2661–2671.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778.

Howard, A. G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; and Adam, H. 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv:1704.04861.

Neftci, E. O.; Mostafa, H.; and Zenke, F. 2019. Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks. *IEEE Signal Processing Magazine*, 36(6): 51–63.

Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; and Chen, L.-C. 2018. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4510–4520.

Xie, S.; Girshick, R.; Dollár, P.; Tu, Z.; and He, K. 2017. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1492–1500.

Zagoruyko, S.; and Komodakis, N. 2016. Wide residual networks. In *Proceedings of the British Machine Vision Conference (BMVC)*.

layers	output size	SEW ResNet8	SEW ResNet14	Wide SEW ResNet8	Wide SEW ResNet14	SEW ResNeXt11	SEW ResNeXt20
conv with bn	112×112	7×7, 64, stride 2					
sn	112×112						
max pool	56×56	3×3, stride 2					
SEW block1	28×28	$\begin{bmatrix}3\times 3, 128\\3\times 3, 128\end{bmatrix}\times 1$	$\begin{bmatrix}3\times 3, 128\\3\times 3, 128\end{bmatrix}\times 2$	$\begin{bmatrix}3\times 3, 128\times 4\\3\times 3, 128\times 4\end{bmatrix}\times 1$	$\begin{bmatrix}3\times 3, 128\times 4\\3\times 3, 128\times 4\end{bmatrix}\times 2$	$\begin{bmatrix}1\times 1, 64\\3\times 3, 64, g=32\\1\times 1, 128\end{bmatrix}\times 1$	$\begin{bmatrix}1\times 1, 64\\3\times 3, 64, g=32\\1\times 1, 128\end{bmatrix}\times 2$
SEW block2	14×14	$\begin{bmatrix}3\times 3, 256\\3\times 3, 256\end{bmatrix}\times 1$	$\begin{bmatrix}3\times 3, 256\\3\times 3, 256\end{bmatrix}\times 2$	$\begin{bmatrix}3\times 3, 256\times 4\\3\times 3, 256\times 4\end{bmatrix}\times 1$	$\begin{bmatrix}3\times 3, 256\times 4\\3\times 3, 256\times 4\end{bmatrix}\times 2$	$\begin{bmatrix}1\times 1, 128\\3\times 3, 128, g=32\\1\times 1, 256\end{bmatrix}\times 1$	$\begin{bmatrix}1\times 1, 128\\3\times 3, 128, g=32\\1\times 1, 256\end{bmatrix}\times 2$
SEW block3	7×7	$\begin{bmatrix}3\times 3, 512\\3\times 3, 512\end{bmatrix}\times 1$	$\begin{bmatrix}3\times 3, 512\\3\times 3, 512\end{bmatrix}\times 2$	$\begin{bmatrix}3\times 3, 512\times 4\\3\times 3, 512\times 4\end{bmatrix}\times 1$	$\begin{bmatrix}3\times 3, 512\times 4\\3\times 3, 512\times 4\end{bmatrix}\times 2$	$\begin{bmatrix}1\times 1, 256\\3\times 3, 256, g=32\\1\times 1, 512\end{bmatrix}\times 1$	$\begin{bmatrix}1\times 1, 256\\3\times 3, 256, g=32\\1\times 1, 512\end{bmatrix}\times 2$
global average pool	1×1	7×7					
fc	1000						

Table 1: Architectures of SNNs. ”sn” denotes the spiking neuron. ”$g = 32$” denotes the grouped convolutions with 32 groups. The hyper-parameters of the spike-element-wise block are shown in the brackets with the number of stacked blocks outside.

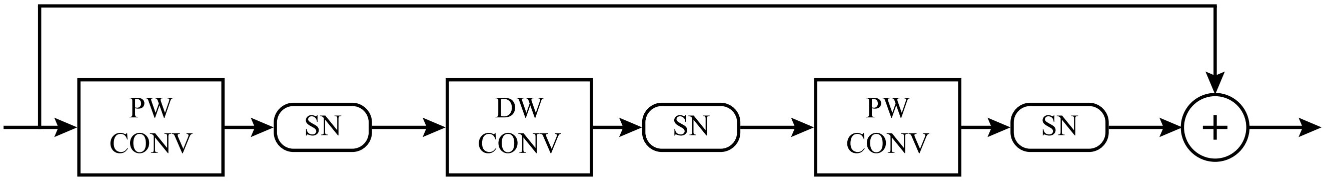


Figure 1: The basic block of SpikingMobileNet. ”PW CONV” is the pointwise convolution and ”DW CONV” is the depthwise convolution. ”SN” is the spiking neuron.

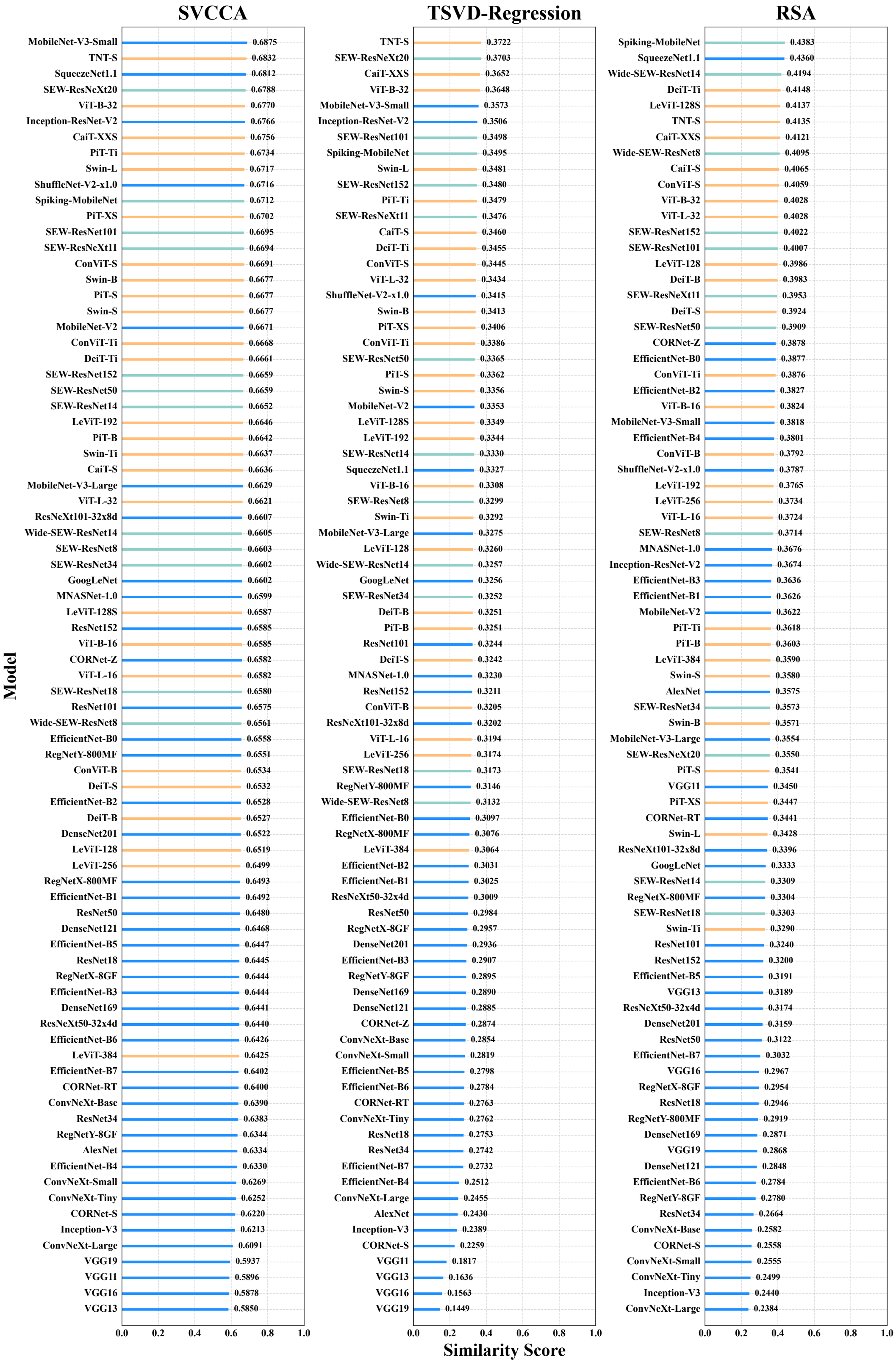


Figure 2: Overall model rankings of the similarity scores on Allen Brain mouse dataset. The similarity scores of CNNs, SNNs and vision transformers are shown by blue, green and orange bars, respectively.

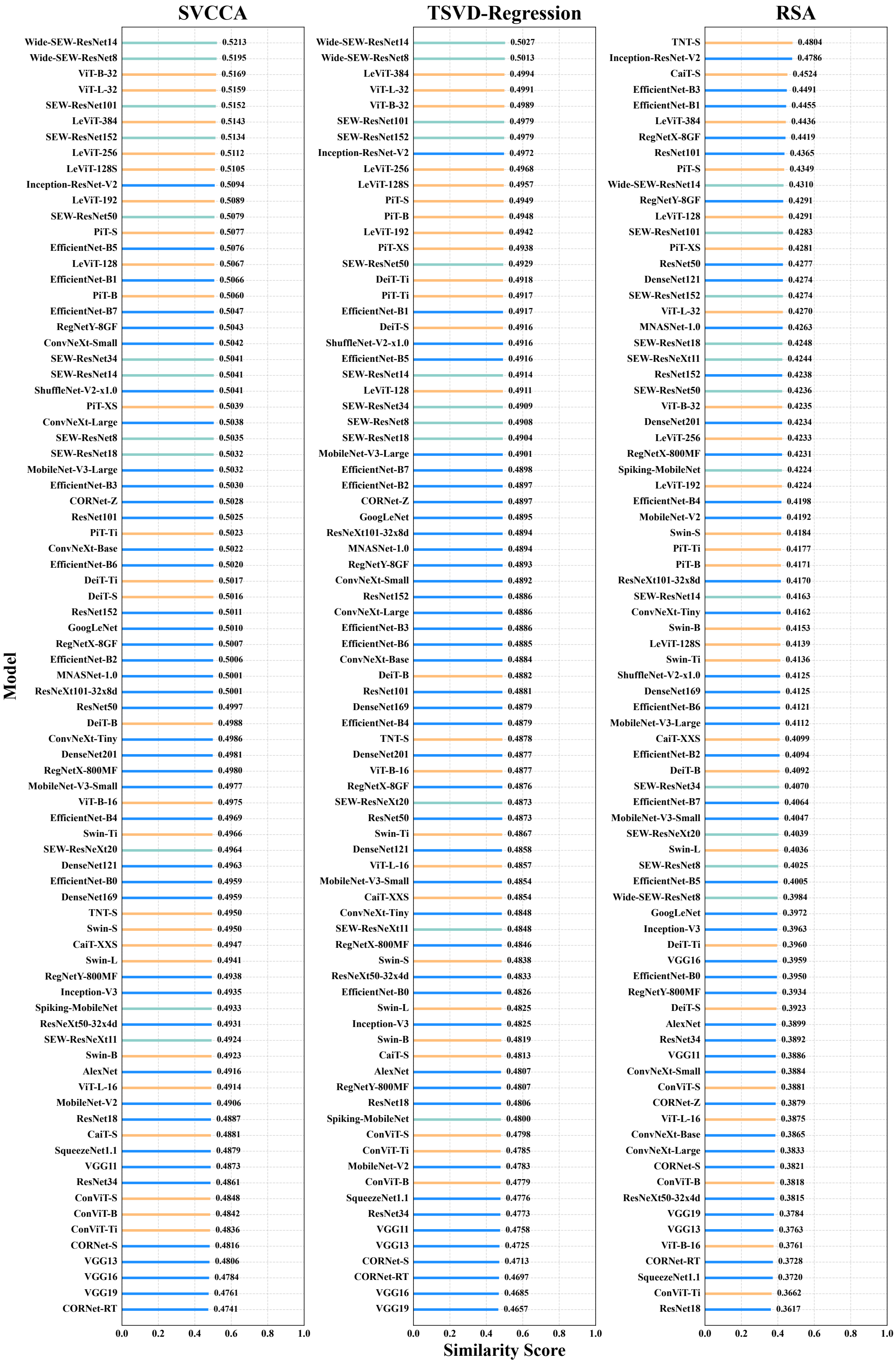


Figure 3: Overall model rankings of the similarity scores on Macaque-Face dataset.

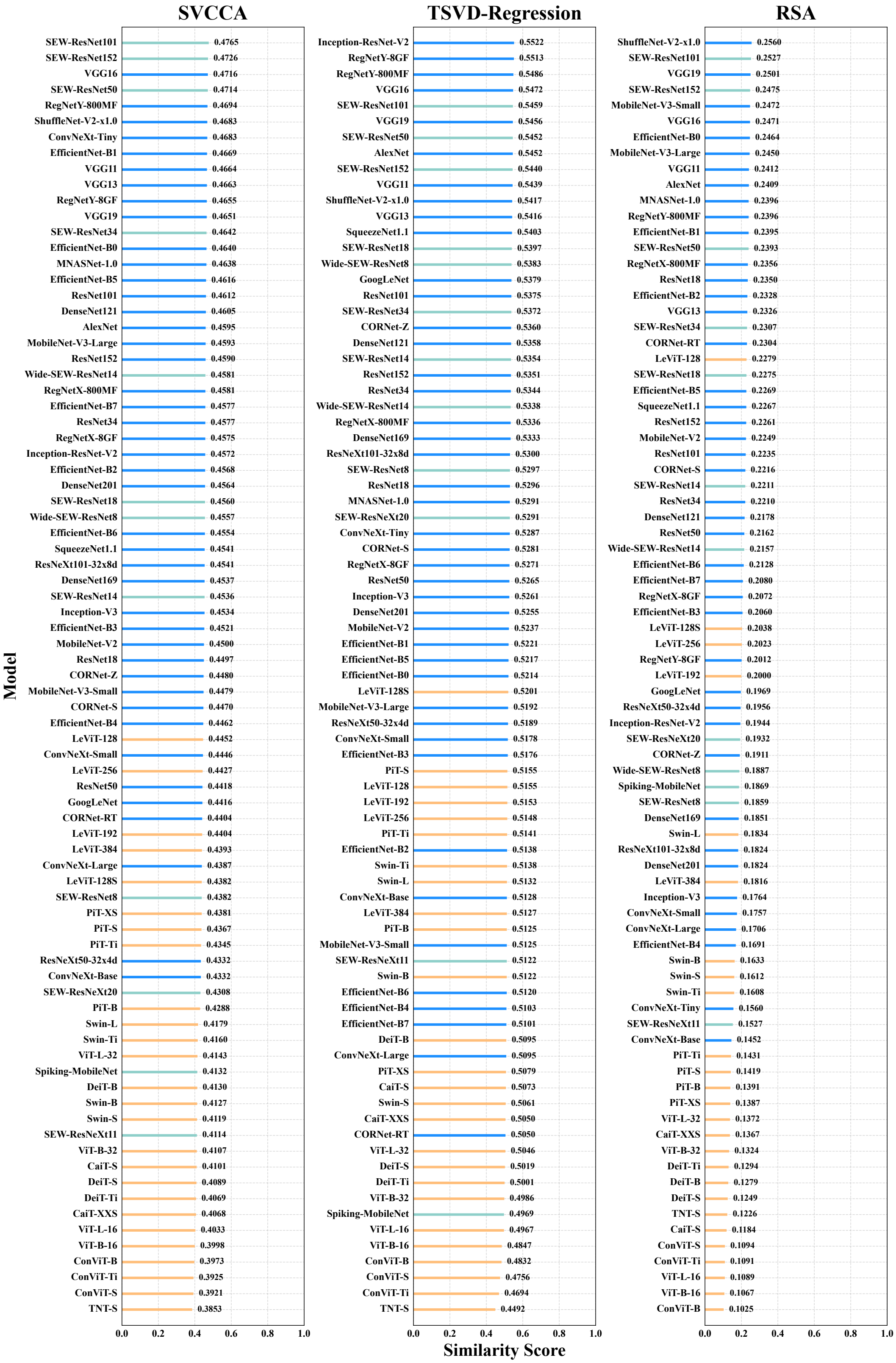


Figure 4: Overall model rankings of the similarity scores on Macaque-Synthetic dataset.

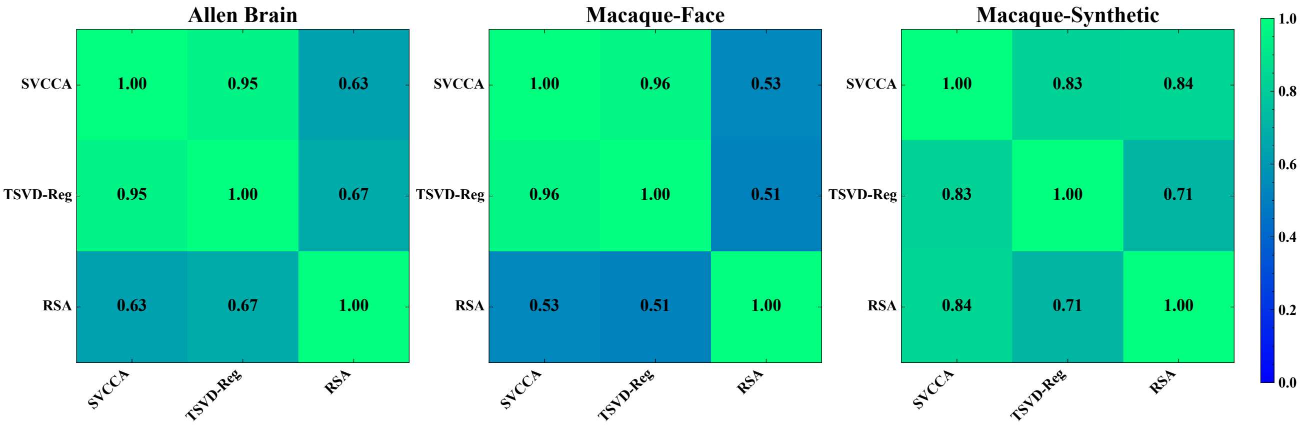


Figure 5: The Spearman's rank correlation between the overall model rankings of different metrics. There is a strong correlation between SVCCA and TSVD-Reg, but RSA has weaker correlations with them.

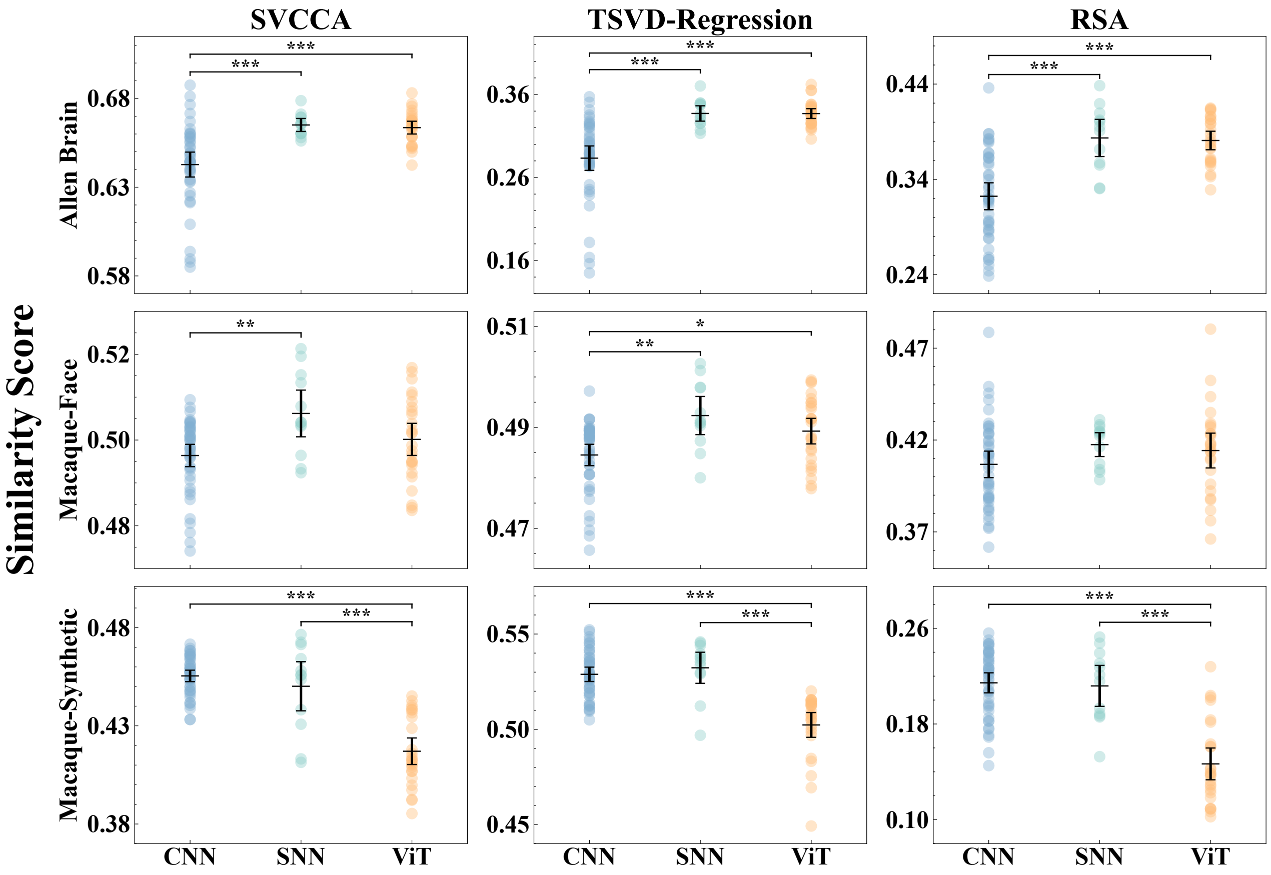


Figure 6: For three datasets and three similarity metrics, each point indicates the final representation similarity score of a model. There are three types of models (CNNs, SNNs, and vision transformers). From top to bottom, each row respectively corresponds to Allen Brain dataset, Macaque-Face dataset, and Macaque-Synthetic dataset. The error bar is the 95% confidence interval across models from a group. The '*' on the horizontal brackets indicates how significant the differences are.

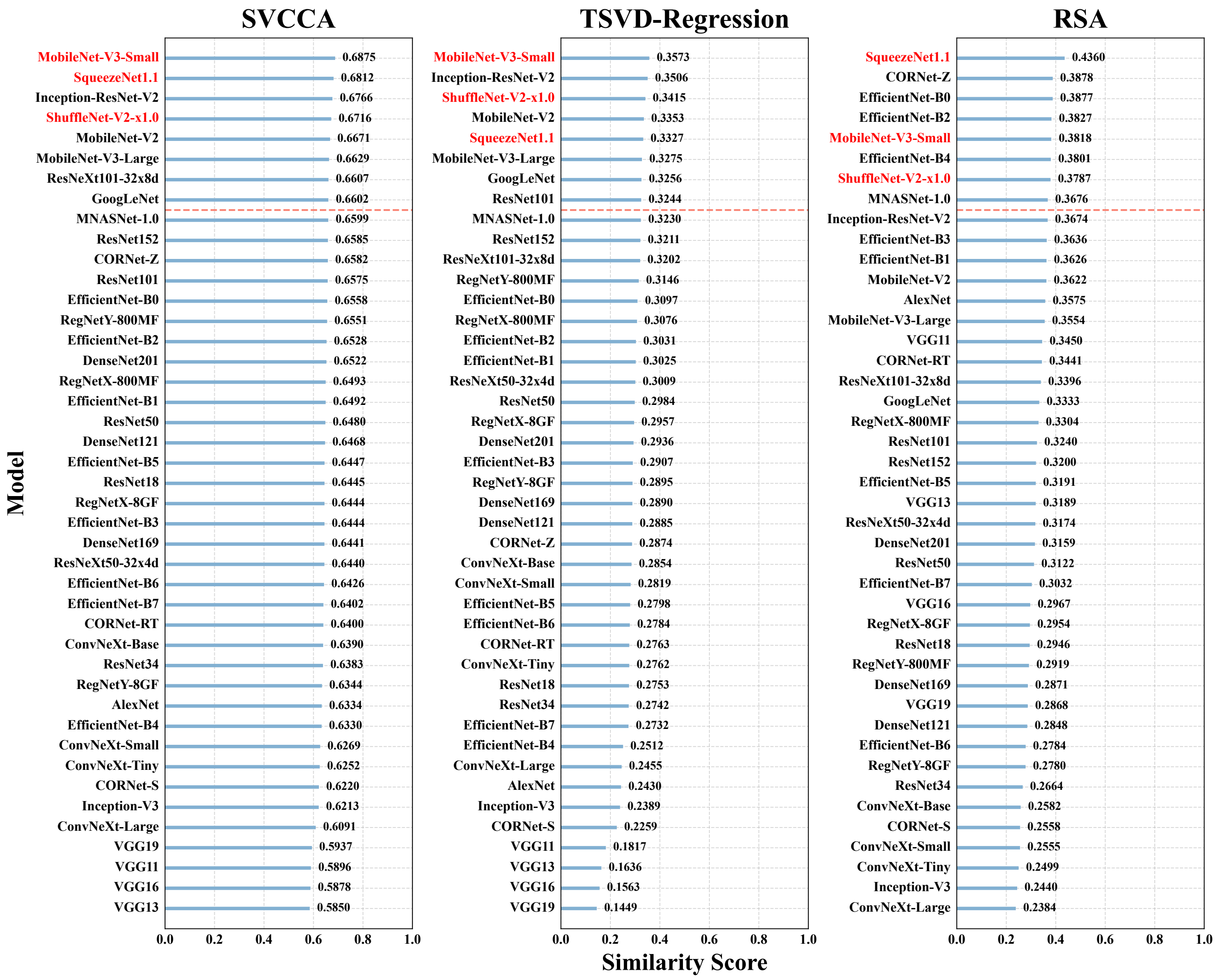


Figure 7: Overall CNN rankings of the similarity scores on Allen Brain mouse dataset. The top 20% models are above the red dotted lines. The networks that are in the top 20% for all three metrics are marked in red.

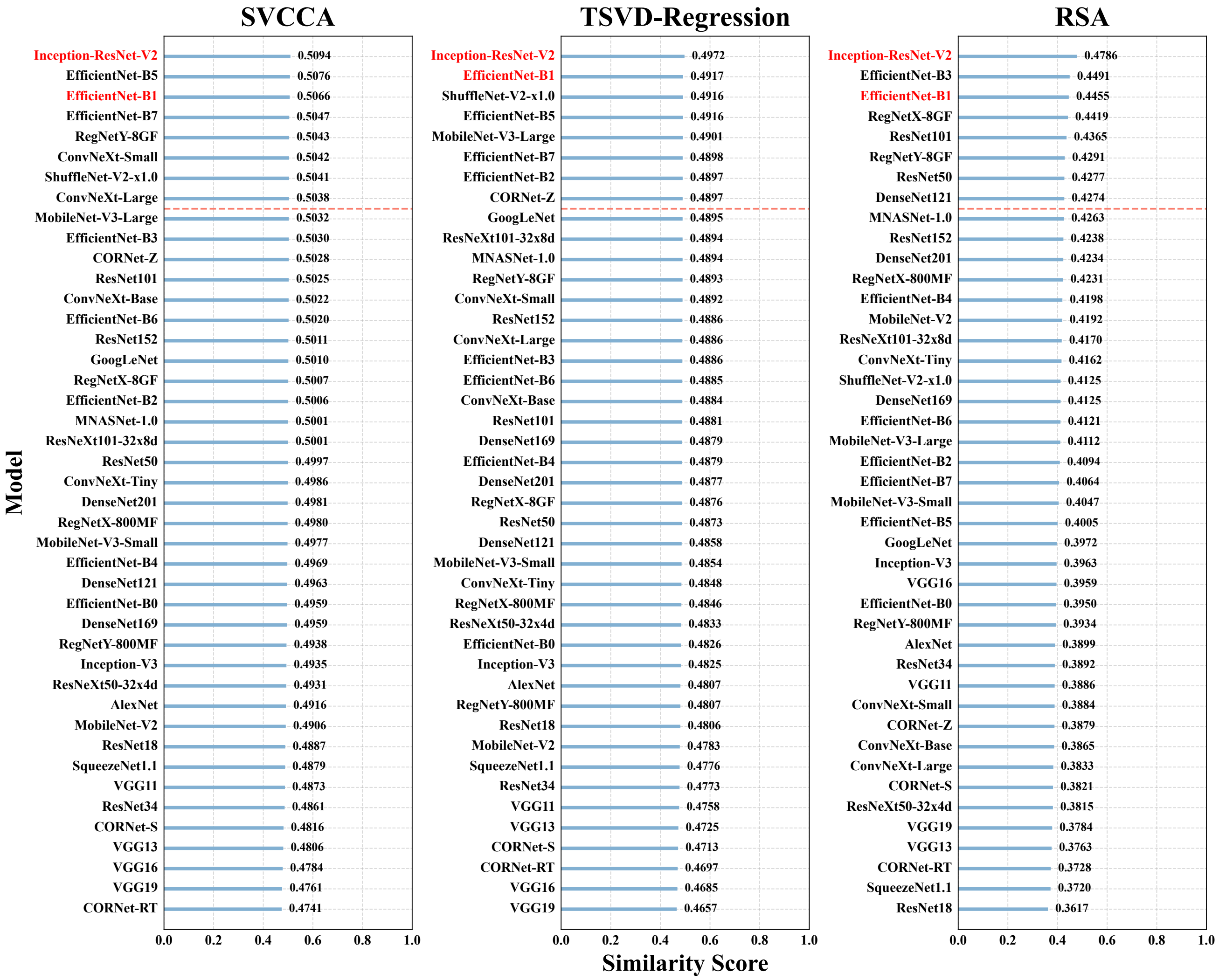


Figure 8: Overall CNN rankings of the similarity scores on Macaque-Face dataset.

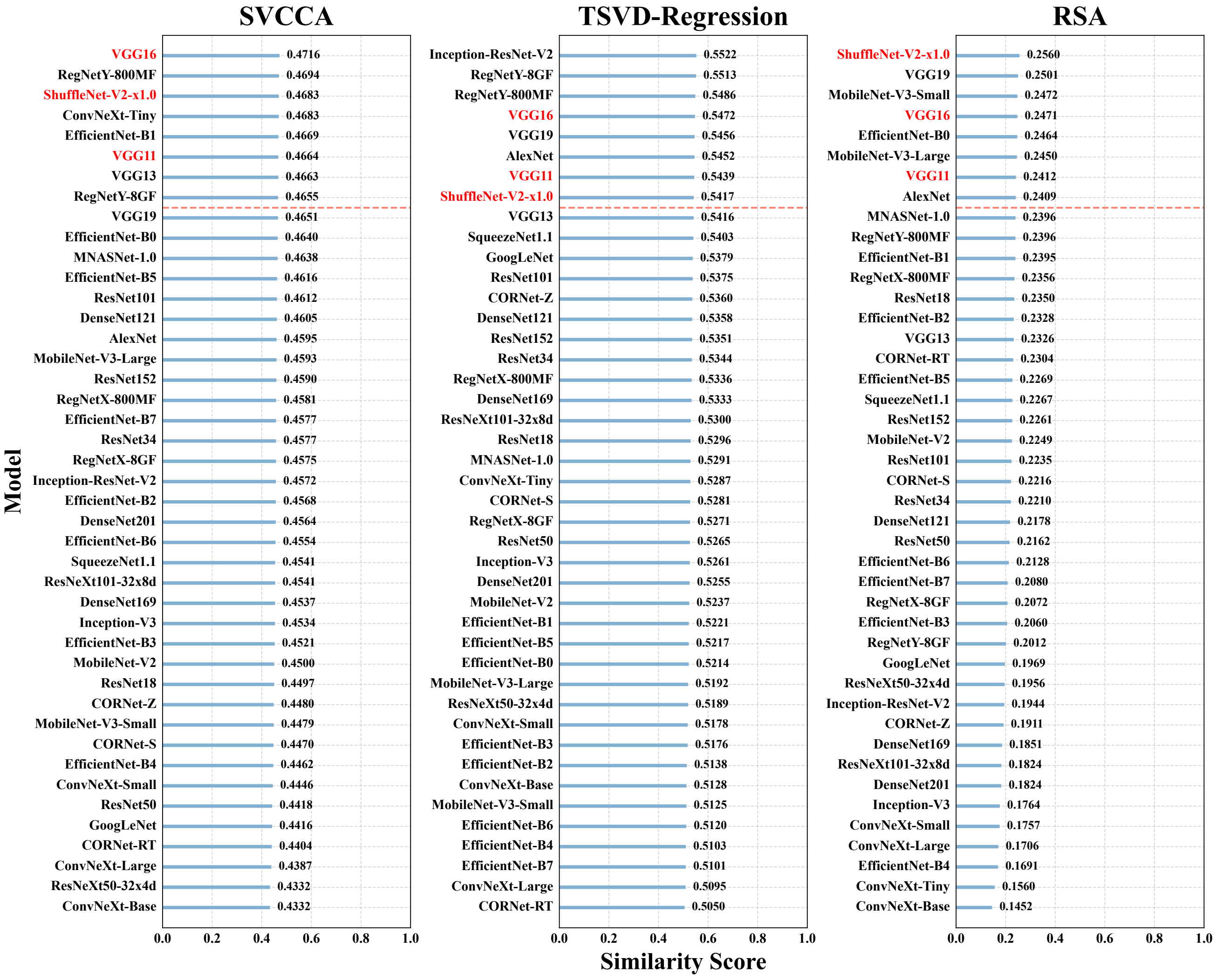


Figure 9: Overall CNN rankings of the similarity scores on Macaque-Synthetic dataset.

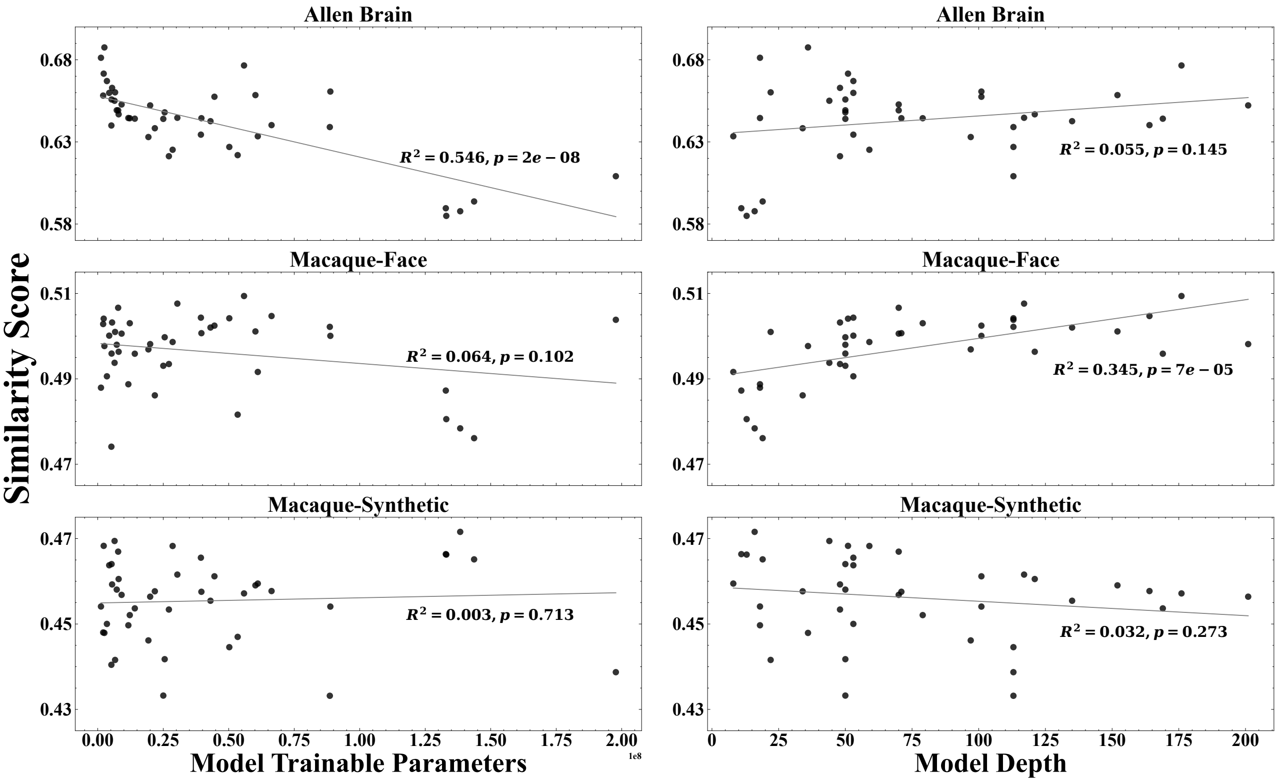


Figure 10: The results of linear regression to model sizes and the similarity scores for SVCCA. Each point indicates a model.

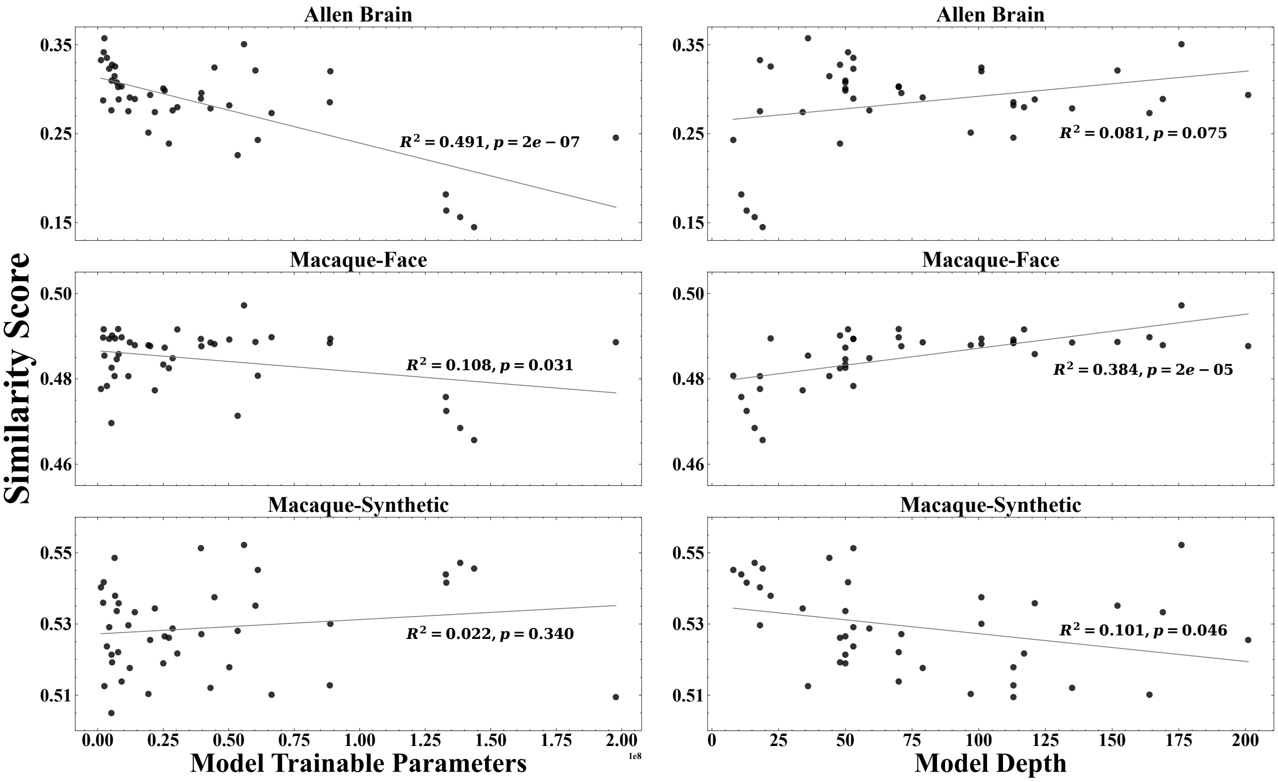


Figure 11: The results of linear regression to model sizes and the similarity scores for TSVD-Reg.

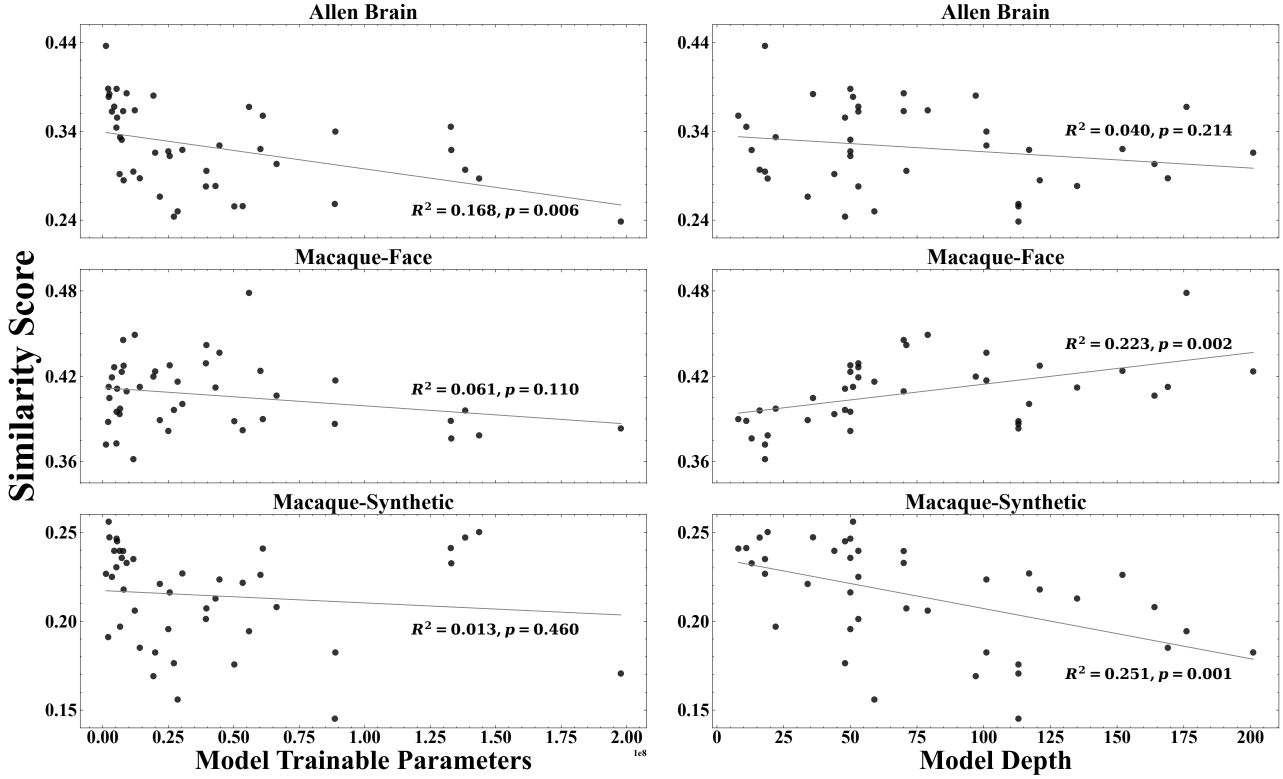
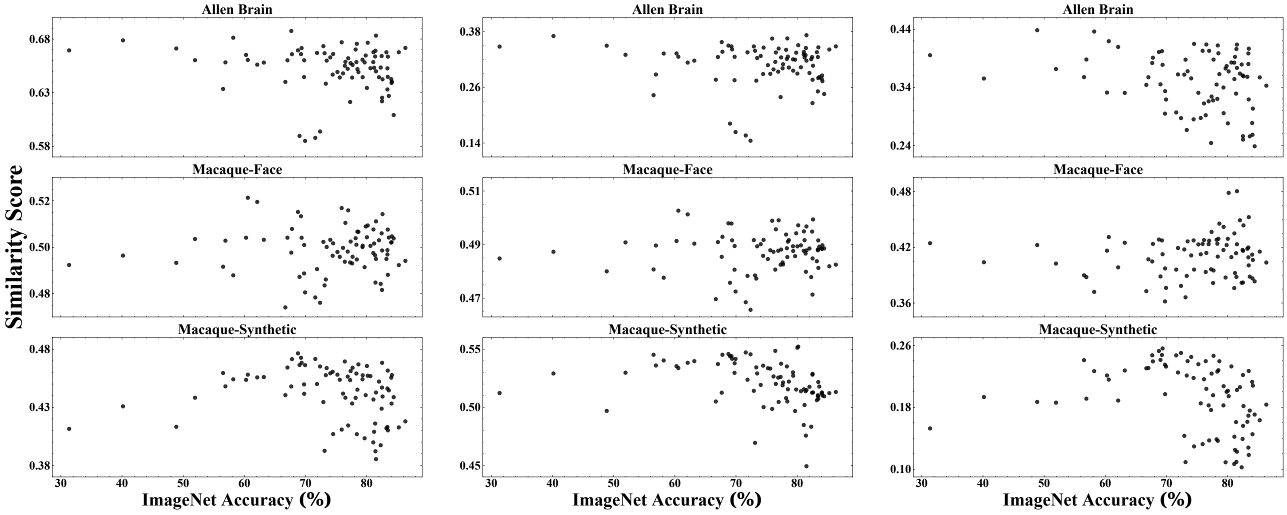


Figure 12: The results of linear regression to model sizes and the similarity scores for RSA.



(a) SVCCA

(b) TSVD-Reg

(c) RSA

Figure 13: The ImageNet accuracy and the similarity scores for SVCCA, TSVD-Reg and RSA.