

VessShape: Few-shot 2D blood vessel segmentation by leveraging shape priors from synthetic images

Wesley Nogueira Galvão¹ and Cesar H. Comin^{1,*}

¹Department of Computer Science, Federal University of São Carlos, São Carlos, SP, Brazil
(Dated: September 26, 2025)

??

Keywords: Blood vessel segmentation, Connectivity, post-processing

I. METODOLOGIA

A. O conjunto de dados VessShape

A geometria das imagens sintéticas é baseada em curvas de Bézier, que permitem uma representação flexível e controlada de formas tubulares. Cada segmento vascular é descrito por uma curva de Bézier de ordem n com pontos de controle $\{\mathbf{p}_i\}_{i=0}^n$. A tortuosidade dos segmentos é ajustada por pequenas perturbações nos pontos de controle, garantindo que a geometria seja realista e diversificada. A curva de Bézier $\mathbf{c}(t)$ de um segmento é dada por

$$\mathbf{c}(t) = \sum_{i=0}^n \binom{n}{i} (1-t)^{n-i} t^i \mathbf{p}_i, \quad (1)$$

onde $t \in [0, 1]$.

Para gerar uma curva, o primeiro (\mathbf{p}_0) e o último (\mathbf{p}_n) pontos de controle são sorteados com distribuição uniforme. Os demais $n - 2$ pontos são inicialmente colocados de forma equiespaçada no segmento que liga \mathbf{p}_0 a \mathbf{p}_n e depois deslocados por uma quantidade aleatória ao longo de um vetor normal unitário. O deslocamento é amostrado de $[-\delta, \delta]$. Valores menores de δ produzem segmentos mais retilíneos.

Para formar a máscara binária M , cada curva é amostrada em resolução suficiente e os pontos sucessivos são conectados formando uma polilinha de 1 pixel. Em seguida aplica-se dilatação morfológica com elemento estruturante disco de raio r_0 , conferindo espessura tubular constante inicial antes de eventuais variações suaves.

Os parâmetros (número de segmentos K , ordem n , escala de deslocamento δ e raio r_0) são sorteados de intervalos predefinidos para maximizar a variabilidade geométrica. A Tabela ?? resume esses parâmetros.

Para compor a imagem final I a partir da máscara M , escolhem-se aleatoriamente duas texturas de classes distintas do ImageNet [?]: uma de primeiro plano F e uma de fundo B . Ambas são recortadas e redimensionadas para $H \times W$. Gera-se um matte A aplicando filtro Gaussiano de desvio σ sobre M e normalizando para $[0, 1]$. A composição é

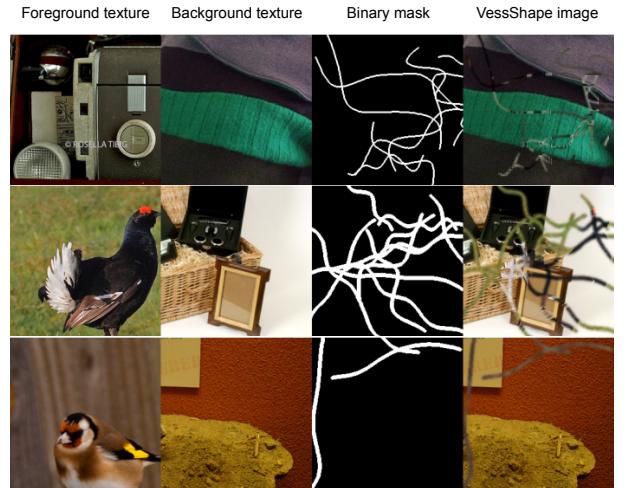


FIG. 1. Exemplos de texturas do ImageNet, máscaras geométricas sintéticas e imagens resultantes VessShape.

$$I = AF + (1 - A)B, \quad (2)$$

de modo que regiões de vaso ($A \approx 1$) preservam F , enquanto o restante ($A \approx 0$) preserva B . O parâmetro σ controla a suavidade das bordas. Após a composição, aplica-se normalização canal a canal usando estatísticas do ImageNet. Exemplos aparecem na Figura ??.

B. Dados reais para validação

Utilizamos os conjuntos DRIVE e VessMAP para avaliar a utilidade dos vieses de forma. DRIVE (40 imagens 584×565 , 20 treino / 20 teste) é referência para vasos de retina. VessMAP (100 imagens 256×256 de microscopia de fluorescência cortical) apresenta ruído e contraste heterogêneos, múltiplos calibres e artefatos. As modalidades diferem: em VessMAP vasos brilhantes em fundo escuro; em DRIVE vasos escuros sobre fundo claro. Exemplos na Figura ??.

* Corresponding author: comin@ufscar.br

TABLE I. Principais parâmetros usados na geração do VessShape.

extbf{Parâmetro}	Intervalo	Descrição
Número de curvas K	[1, 20]	Quantidade de ramos/vasos por amostra.
Pontos de controle $n+1$	[2, 20]	Complexidade (ordem n) da curva de Bézier.
Escala de deslocamento δ (px)	[50.0, 150.0]	Amplitude típica do deslocamento dos pontos de controle (tortuosidade).
Raio inicial r_0 (px)	[1, 5]	Espessura basal; aplica-se afilamento suave ao longo do ramo.
Desfoco de matting σ	[1, 2]	Desvio padrão da Gaussiana para $A = G_\sigma * M$.

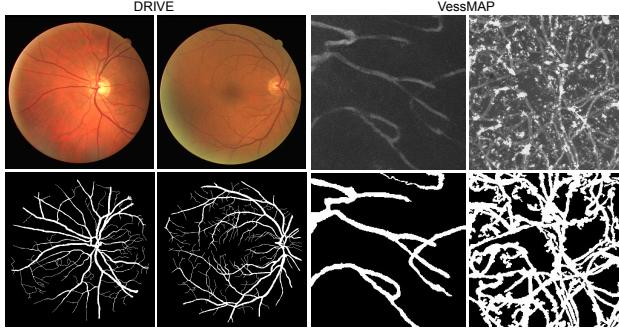


FIG. 2. Exemplos das bases DRIVE e VessMAP e respectivas máscaras anotadas.

TABLE II. Hiperparâmetros de pré-treinamento no VessShape.

extbf{Hiperparâmetro}	VSUNet50	VSUNet18
Batch size	96	192
Learning rate	1.0e-3	1.0e-2
LR decay	0.0	0.0
Weight decay	1.0e-4	0.0
Imgs/epoch	50,000	50,000
Max epochs	3000	1000

C. Arquitetura e estratégias de treinamento

Adotamos arquitetura U-Net encoder-decoder com conexões de atalho. Comparamos variantes com encoders ResNet18 e ResNet50 [?], instanciadas via *Segmentation Models Pytorch*¹.

Dois cenários: (i) treinamento do zero em DRIVE e VessMAP; (ii) pré-treinamento em VessShape seguido de fine-tuning. Assim, distinguimos três procedimentos: (a) training from scratch; (b) pre-training em VessShape; (c) fine-tuning nos dados reais.

D. Pré-treinamento em VessShape

O pré-treinamento busca expor o modelo a vasta diversidade geométrica com texturas variáveis como pista secundária, favorecendo recuperação de estruturas finas.

TABLE III. Desempenho das variantes VSUNet após pré-treinamento no VessShape.

extbf{Métrica}	VSUNet50	VSUNet18
Dice	0.861 ± 0.022	0.859 ± 0.077
Acc	0.960 ± 0.008	0.956 ± 0.037
IoU	0.758 ± 0.032	0.761 ± 0.096
Prec	0.780 ± 0.037	0.774 ± 0.096
Rec	0.964 ± 0.012	0.974 ± 0.018

Treinamos dois modelos (VSUNet18 e VSUNet50). A Tabela ?? resume hiperparâmetros.

E. Fine-tuning

Após o pré-treinamento, os pesos são ajustados em subsets de DRIVE e VessMAP (cenários few-shot e full) para avaliar ganho de amostragem e transferência de forma.

Para gerar cada máscara binária, o número de segmentos K , a ordem n das curvas de Bézier, a escala de deslocamento

δ e o raio r_0 são todos amostrados aleatoriamente de um intervalo para garantir uma ampla variedade de formas. A Tabela eftab:vessshape_paramsresume os parâmetros usados na geração do con-

Para compor a imagem final I a partir de uma máscara binária M , uma textura de primeiro plano F e uma textura de fundo B são inseridas, respectivamente, nos segmentos de vasos gerados e no fundo da imagem. As texturas são selecionadas aleatoriamente do ImageNet citeJiaDeng2009. Especificamente, para cada máscara M , duas imagens são sorteadas aleatoriamente de duas classes distintas do conjunto de dados ImageNet. As imagens são então cortadas e redimensionadas aleatoriamente para as dimensões alvo ($H \times W$). Uma máscara alfa A é então gerada suavizando M com um filtro Gaussiano de desvio padrão

σ e normalizando seus valores para o intervalo $[0, 1]$. As texturas são subsequentemente misturadas usando esta máscara de acordo com

$$I = A F + (1 - A) B, \quad (3)$$

o que garante que as regiões dos vasos ($A \approx 1$) preservem o primeiro plano, enquanto as regiões não vasculares ($A \approx 0$) retenham o fundo.

¹ https://github.com/qubvel/segmentation_models.pytorch

TABLE IV. Principais parâmetros usados para gerar o conjunto de dados VessShape.

Parâmetro	Intervalo	Descrição
Número de curvas K	[1, 20]	Número de ramos/vasos gerados por amostra. Pontos de controle $n+1$
[2, 20] delta(px)	Complexidade da curva de Bézier (ordem n). Escala de deslocamento [50.0, 150.0]	Controla a curvatura/tortuosidad através da amplitude típica do deslocamento do ponto de controle. Raio inicial r_0 (px)
[1, 5] sigma	Espessura basal do vaso; um afilamento suave é aplicado ao longo do ramo. Borrão de matting [1, 2]	Desvio padrão da Gaussiana usada para $A = G$
sigma * M.		

O parâmetro

σ controla a suavidade das bordas do vaso. Após a composição, a imagem I passa por uma normalização por canal usando as estatísticas do ImageNet para compatibilidade com modelos pré-treinados. Exemplos de máscaras e imagens geradas são mostrados na Figura `eff:vessshapeample`.

F. Dados do mundo real para validação

Para quantificar a utilidade do viés de forma introduzido pelo conjunto de dados VessShape, consideraremos dois conjuntos de dados de vasos sanguíneos: DRIVE e VessMAP. O conjunto de dados DRIVE cite serve como um padrão popular para benchmarking de algoritmos de segmentação de vasos da retina e é composto por 40 fotografias de fundo de olho divididas em 20 para

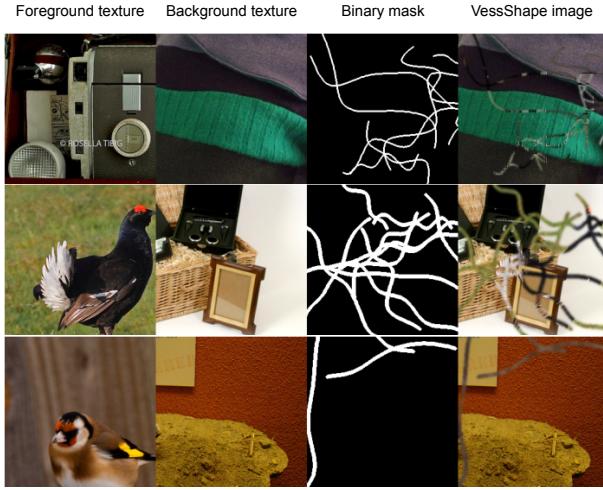


FIG. 3. Exemplos de amostras do ImageNet, geometria sintética e respectivas imagens VessShape. As amostras do ImageNet são usadas como texturas para as máscaras binárias sintéticas, definindo a respectiva imagem VessShape.

treinamento e 20 para teste, cada uma medindo 584×565 pixels. O conjunto de dados VessMAP consiste em 100 imagens, de 256×256 pixels cada, adquiridas por microscopia de fluorescência do córtex do camundongo. Ele foi curado para incluir uma variedade de características vasculares desafiadoras, como ruído e níveis de contraste inconsistentes, diferentes tamanhos de vasos, artefatos de imagem proeminentes e flutuações de intensidade dentro das estruturas vasculares.

Os dois conjuntos de dados originam-se de modalidades de imagem fundamentalmente diferentes, resultando em características distintas. As imagens de fundo de olho no DRIVE, que capturaram toda a retina, possuem uma estrutura global clara que inclui marcos como o disco óptico. As amostras também contêm muitos vasos muito finos que são difíceis de segmentar. Em contraste, as imagens do VessMAP são visualizações altamente ampliadas de pequenas áreas corticais e não possuem organização global discernível. As bordas dos vasos são geralmente menos definidas do que os vasos do DRIVE. Outra diferença fundamental é que, sem qualquer processamento, os vasos no VessMAP são brilhantes com fundos escuros, enquanto os vasos no DRIVE são escuros com fundos brilhantes. A Figura `eff:drive_vessmapsamples` mostra uma amostra de cada conjunto de dados.

G. Arquiteturas de modelo e estratégias de treinamento

Adotamos um design U-Net codificador-decodificador, ou seja, o decodificador é simétrico ao codificador e o modelo contém conexões de atalho entre diferentes estágios codificador-decodificador. Dois modelos são com-

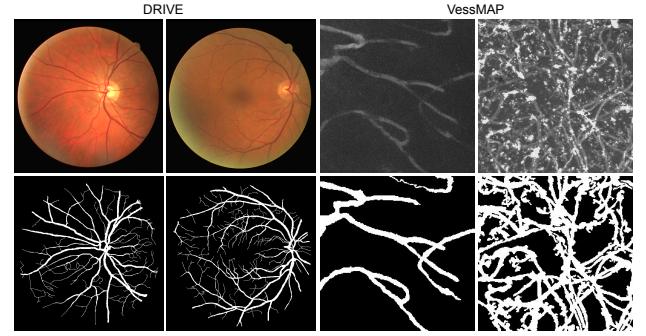


FIG. 4. Amostras dos conjuntos de dados DRIVE e VessMAP e suas respectivas máscaras de verdade do terreno.

parados, um com um codificador ResNet18 e outro com um codificador ResNet50 `citehe2016deep`. Os modelos foram instanciados a partir do pacote Python `extitSegmentation Models Pytorch` https://github.com/qubvel/segmentation_models.pytorch.

Dois cenários de treinamento são considerados. No primeiro, o treinamento é feito do zero separadamente nos conjuntos de dados DRIVE e VessMAP para estabelecer uma linha de base. O segundo cenário consiste em pré-treinar no conjunto de dados VessShape e ajustar fino no DRIVE e VessMAP para medir a transferibilidade e a eficiência da amostra das representações aprendidas. Esses dois cenários envolvem três procedimentos de treinamento distintos: i) treinamento do zero em conjuntos de dados naturais; ii) pré-treinamento no VessShape e iii) ajuste fino em dados do mundo real. Esses procedimentos são descritos a seguir.

H. Treinamento do zero em conjuntos de dados naturais

I. Pré-treinamento no VessShape

O pré-treinamento no VessShape visa expor o modelo a uma ampla variedade de geometrias tubulares, mantendo a textura como uma pista secundária, o que tende a beneficiar a segmentação de estruturas finas. O modelo é treinado para minimizar a perda média sobre um grande número de amostras extraídas do conjunto de dados sintético.

Concretamente, pré-treinamos dois modelos U-Net, um com um codificador ResNet18 e outro com um codificador ResNet50. Nós nos referimos a eles como VSUNet18 e VSUNet50, respectivamente. O fluxo de treinamento expõe intencionalmente o modelo a geometrias do tipo vaso, enquanto as texturas continuam mudando no fundo. Essa narrativa de formas abundantes e aparências mutáveis leva a rede a confiar menos em pistas de textura superficiais e mais em regularidades geométricas que importam para a conectividade e a recuperação de estruturas finas. Abaixo, detalhamos a

configuração do VSUNet50 e resumimos seu desempenho no VessShape.

J. Ajuste fino em conjuntos de dados naturais

- Conjuntos de dados (VessMAP e DRIVE)
- Procedimento

II. RESULTADOS

III. CONCLUSÃO

FUNDING

C. H. Comin thanks FAPESP (grant no. 21/12354-8) for financial support.

□ Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2009) pp. 248–255.

□ 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* (2016) pp. 770–778.

TABLE V. Hiperparâmetros de pré-treinamento no VessShape.

Hiperparâmetro	VSUNet50	VSUNet18
Tamanho do lote	96	192
Taxa de aprendizado	1.0e-3	1.0e-2
Decaimento da taxa de aprendizado	0.0	0.0
Decaimento de peso	1.0e-4	0.0
Imgs/época	50, !000	50, !000
Épocas máximas	3000	1000

TABLE VI. Desempenho das variantes VSUNet após pré-treinamento no VessShape.

Métrica	VSUNet50	VSUNet18
Dice	0.861 ± 0.022	0.859 ± 0.077
Acc	0.960 ± 0.008	0.956 ± 0.037
IoU	0.758 ± 0.032	0.761 ± 0.096
Prec	0.780 ± 0.037	0.774 ± 0.096
Rec	0.964 ± 0.012	0.974 ± 0.018

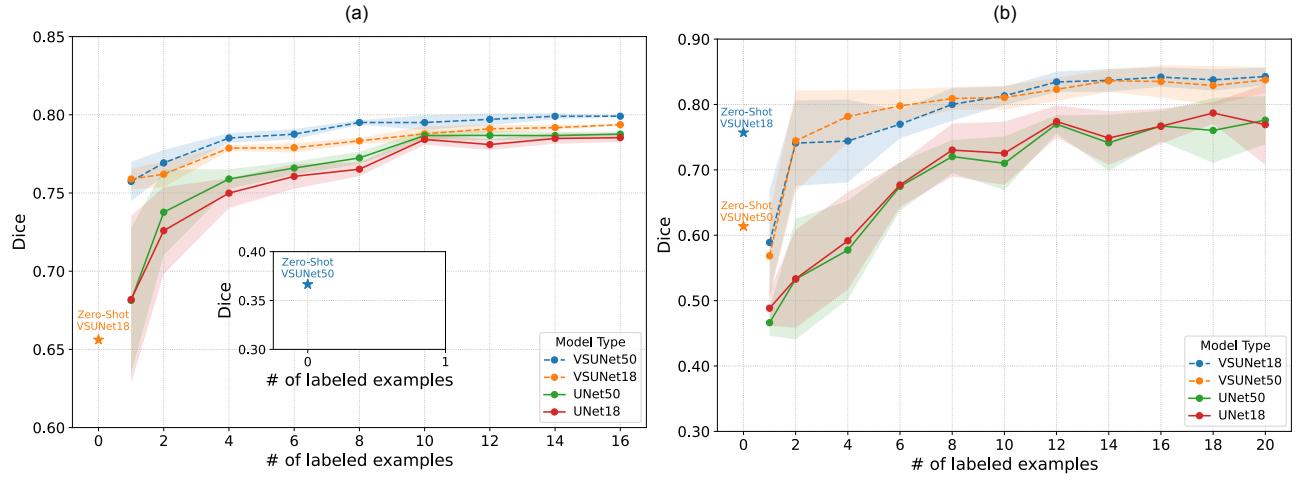


FIG. 5. ??.

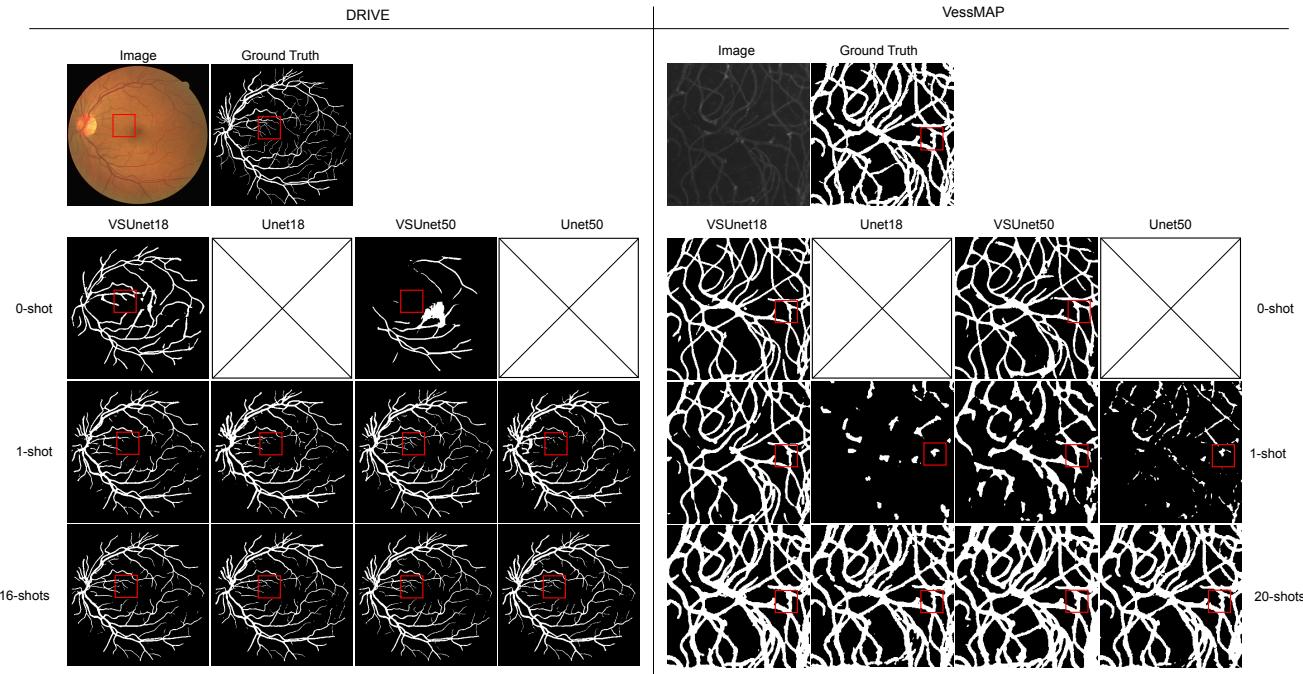


FIG. 6. ??.

TABLE VII. Segmentação few-shot e zero-shot em VessMAP e DRIVE. Os valores são média \pm desvio padrão sobre execuções repetidas (ajuste fino para modelos VSUNet e treinamento do zero para baselines U-Net) avaliados em cada conjunto de teste do conjunto de dados. As linhas de zero-shot vêm de uma única inferência pré-treinada (sem desvio disponível).

Conjunto de dados	#Exemplos	Modelo	Dice	Acc	IoU	Prec	Rec
DRIVE	0	VSUNet18	0.656 \pm 0.000	0.907 \pm 0.000	0.490 \pm 0.000	0.629 \pm 0.000	0.699 \pm 0.000
		VSUNet50	0.367 \pm 0.000	0.888 \pm 0.000	0.230 \pm 0.000	0.728 \pm 0.000	0.275 \pm 0.000
	1	VSUNet18	0.759 \pm 0.007	0.941 \pm 0.002	0.612 \pm 0.009	0.787 \pm 0.021	0.741 \pm 0.019
		VSUNet50	0.757 \pm 0.013	0.939 \pm 0.006	0.611 \pm 0.016	0.773 \pm 0.046	0.754 \pm 0.039
	16	UNet18	0.682 \pm 0.054	0.919 \pm 0.016	0.523 \pm 0.058	0.717 \pm 0.079	0.690 \pm 0.118
		UNet50	0.681 \pm 0.046	0.916 \pm 0.031	0.523 \pm 0.050	0.722 \pm 0.099	0.690 \pm 0.110
	16	VSUNet18	0.794 \pm 0.000	0.950 \pm 0.000	0.658 \pm 0.000	0.833 \pm 0.004	0.762 \pm 0.003
		VSUNet50	0.799 \pm 0.001	0.952 \pm 0.000	0.666 \pm 0.001	0.846 \pm 0.003	0.762 \pm 0.004
		UNet18	0.785 \pm 0.003	0.946 \pm 0.001	0.647 \pm 0.004	0.795 \pm 0.005	0.781 \pm 0.004
		UNet50	0.788 \pm 0.002	0.947 \pm 0.001	0.650 \pm 0.002	0.807 \pm 0.006	0.774 \pm 0.004
VessMAP	0	VSUNet18	0.757 \pm 0.000	0.886 \pm 0.000	0.616 \pm 0.000	0.846 \pm 0.000	0.696 \pm 0.000
		VSUNet50	0.614 \pm 0.000	0.817 \pm 0.000	0.472 \pm 0.000	0.746 \pm 0.000	0.605 \pm 0.000
	1	VSUNet18	0.589 \pm 0.080	0.705 \pm 0.205	0.455 \pm 0.088	0.675 \pm 0.205	0.738 \pm 0.148
		VSUNet50	0.569 \pm 0.063	0.708 \pm 0.203	0.439 \pm 0.071	0.683 \pm 0.208	0.700 \pm 0.167
	20	UNet18	0.488 \pm 0.027	0.674 \pm 0.184	0.362 \pm 0.032	0.682 \pm 0.204	0.622 \pm 0.210
		UNet50	0.466 \pm 0.020	0.665 \pm 0.181	0.343 \pm 0.025	0.674 \pm 0.206	0.604 \pm 0.215
	20	VSUNet18	0.843 \pm 0.013	0.901 \pm 0.021	0.739 \pm 0.018	0.825 \pm 0.049	0.888 \pm 0.048
		VSUNet50	0.837 \pm 0.020	0.905 \pm 0.025	0.732 \pm 0.026	0.851 \pm 0.032	0.850 \pm 0.027