Supplementary Information for A Robust End-to-end Method for Parametric Curve Tracing via Soft Cosine-similarity-based Objective Function

Boran Han, Jeremy Vila
Shell International Exploration and Production Inc.
2 Drydock Ave., Boston MA, 02210
Boran.Han@shell.com, Jeremy.Vila@shell.com

1. Objective function without B-spline kernel is not differentiable.

In this section, we prove that cosine similarity loss (Eq. 10 in main text) is not differentiable with respect to parameter $\boldsymbol{\beta}$ when P_t is a vector constituted by nearby pixel intensities (Eq. 9 in main text) due to the discrete indexing by $\boldsymbol{\beta}$.

$$\frac{\partial L_{\text{corr}}}{\partial \boldsymbol{\beta}} = \frac{\partial \sum_{t=0}^{\Delta t} (f_{\text{corr}}(\boldsymbol{P}_t, \boldsymbol{P}_{t+1})}{\partial (\boldsymbol{P}_t, \boldsymbol{P}_{t+1})} \frac{\partial (\boldsymbol{P}_t, \boldsymbol{P}_{t+1})}{\partial \boldsymbol{\beta}} \tag{1}$$

$$\frac{\partial (\boldsymbol{P}_t, \boldsymbol{P}_{t+1})}{\partial \boldsymbol{\beta}} = \frac{\partial (\boldsymbol{P}_t, \boldsymbol{P}_{t+1})}{\partial Y} \nabla_{\boldsymbol{\beta}} \lfloor Y(\boldsymbol{\beta}) \rfloor$$

$$\nabla_{\boldsymbol{\beta}} \lfloor Y(\boldsymbol{\beta}) \rfloor = \boldsymbol{\gamma} \circ \boldsymbol{\chi} = \begin{bmatrix} \gamma_1 \chi_1 \\ \gamma_2 \chi_2 \\ \vdots \\ \gamma_N \chi_N \end{bmatrix} \tag{2}$$

where polynomial coefficient $N \geq 0$, γ are constant real numbers, χ_n is the indicator function of Y:

$$\chi_n = \begin{cases} 1 & \lfloor Y(\beta_n) \rfloor = Y(\beta_n) \\ 0 & \lfloor Y(\beta_n) \rfloor \neq Y(\beta_n) \end{cases} \tag{3}$$

It indicates that during the optimization, minor change in $\boldsymbol{\beta}$ leads to invariant of $(\boldsymbol{P}_t, \boldsymbol{P}_{t+1})$. And $\nabla_{\boldsymbol{\beta}} \lfloor Y(\boldsymbol{\beta}) \rfloor$ is a step function. it is impossible to calculate the gradients, which completes the proof. We note that $\frac{\partial L_{\text{corr}}}{\partial \boldsymbol{\alpha}}$, which is independent and exchangeable with $\frac{\partial L_{\text{corr}}}{\partial \boldsymbol{\beta}}$, also follow the same behaviors in term of differentiablity.

2. Parameter discussion

We list the choice of parameters for all datasets in Tab. 1. It is worth noting that Gaussian filters are applied before parametric curve tracing when performing the lane following. In contrast, other experiments presented in this paper perform without additional filters. Such a step in this experiment can improve the performance of our method (Tab. 2). It is because that the width of the lane separation lines varies significantly towards the horizon. Adding the Gaussian filter as an additional step can improve the signal similarity along the lane separation lines, which aligns well with the scope of our method described in Sec. 1 in the main text.

Dataset	W	λ_1	λ_2	Δt	N
SPT/SMT	2	0	0	1	1
Lane Following	10	0.01	0.001	3	2
Seismic	3	0	0	Length of the entire curve ~ 25	2

Table 1. Choice of parameters in each dataset. Seismic dataset will be introduced in Sec. 6

Noise level	Low light $\sigma_{est} = 0.335$ [6]		
	$\mathscr{L}$	$\mathscr{D}$	$\mathscr{V}$
Without Gaussian filter	0.407	5.043	0.355
With Gaussian filter	0.823	2.171	0.188

Table 2. Evaluation results of lane following with and without Gaussian filter added before curve tracing

There are four parameters in parametric curve tracing: W, λ_1, λ_2 and Δt. The most important ones are the regularization parameters λ_1 and λ_2, which are used to balance the contribution between the data fidelity and curvature continuity. The larger the λ_2 is, the smoother the reconstruction is. However, in the case of SPT/SMT, smoothness is not required due to frame-to-frame random walk behaviors. In such a case, λ_2 will be set to 0. The positive parameter W and Δt can balance the influence from the surroundings and local features. We observe that the larger W and Δt are, the robust our method will be against the noise. However too large W and Δt can not guarantee stability and can also increase the computational cost. Therefore, one should be choose the parameters properly to preserve the accuracy in low SNR images

while controlling the computational cost. We use low-right lane following as an example showing performance can be affected by difference choice of parameters. Four parameters (W, Δt, λ_1 and λ_2) are changed individually from the optimal choice, as the results shown in Tab. 3 and Tab. 4.

Noise level	Low light $\sigma_{est} = 0.335$ [6]		
Parameters	$\mathscr{L}$	$\mathscr{D}$	$\mathscr{V}$
$W = 5, \Delta t = 3$	0.857	3.322	0.216
$W = 2, \Delta t = 3$	0.695	3.655	0.229
$W = 10, \Delta t = 1$	0.947	1.860	0.253
$W = 10, \Delta t = 5$	0.861	3.519	0.191

Table 3. Evaluation results of lane following using different W and Δt

Noise level	Low light $\sigma_{est} = 0.335$ [6]		
Parameters	$\mathscr{L}$	$\mathscr{D}$	$\mathscr{V}$
$\lambda_1 = 0, \lambda_2 = 0.001$	0.811	3.531	0.190
$\lambda_1 = 0.1, \lambda_2 = 0.001$	0.887	2.595	0.191
$\lambda_1 = 0.01, \lambda_2 = 0$	0.853	2.312	0.192
$\lambda_1 = 0.01, \lambda_2 = 0.1$	0.547	4.251	0.169

Table 4. Evaluation results of lane following using different λ_1 and λ_2

3. Cosine similarity term is Gaussian noise-tolerant

We here show the proof that our method is relatively tolerate to Gaussian noise. As stated in Main Sec. 3, the parametric parameters describing curves, $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$, are obtained by minimizing objective function L. Our objective function is made of three parts, $L = L_{\text{corr}} + \lambda_1 L_{\text{reg}} + \lambda_2 L_{\text{cont}}$. L_{reg} and L_{cont} are irrelevant to pixel intensities $\boldsymbol{P}_t$ as they only describe the internal features of curves. Therefore, in order to prove our objective function is Gaussian noise tolerate, we only need to prove that the set of $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$, corresponding to the minimum L_{corr}, is invariant to noise. Hence:

$$\boldsymbol{\beta}, \boldsymbol{\alpha} = \arg\min L_{\text{corr}} \sim \arg\min L^n_{\text{corr}} \tag{4}$$

where L^n_{corr} are the cosine-similarity objective term when Gaussian noises are presented. Without losing its generality, Gaussian noise $\mathscr{N}_{2W+1}(0,\sigma)$ are added to $\boldsymbol{P}_t$ and $\boldsymbol{P}_{t+1}$ as $\boldsymbol{P}_n \in \mathbb{R}^{2W+1}$ and $\boldsymbol{P}_{n+1}, \in \mathbb{R}^{2W+1}$. .

$$\boldsymbol{P}_n = \boldsymbol{P}_t + \mathscr{N}_{2W+1}(0,\sigma) = \mathscr{N}_{2W+1}(\boldsymbol{P}_t, \sigma) \tag{5}$$

$$\boldsymbol{P}_{n+1} = \boldsymbol{P}_{t+1} + \mathscr{N}_{2W+1}(0,\sigma) = \mathscr{N}_{2W+1}(\boldsymbol{P}_{t+1}, \sigma) \tag{6}$$

where $\mathscr{N}_{2W+1}(\boldsymbol{P}_t, \sigma)$ represents a normal distribution with freedom degree of $2W+1$, mean equal to $\boldsymbol{P}_t$ and variance equal to σ. Our cosine similarity term becomes:

$$L^n_{\text{corr}} = 1 - \frac{\boldsymbol{P}_n \cdot \boldsymbol{P}_{n+1}}{\|\boldsymbol{P}_n\| \|\boldsymbol{P}_{n+1}\|} \tag{7}$$

In order to obtain the behaviors of $\arg\min L^n_{\text{corr}}$ with respect to $\mathscr{N}_{2W+1}(\boldsymbol{P}_t, \sigma)$, we continue to investigate the probability density function of L^n_{corr} as randomness following normal distribution $\mathscr{N}_{2W+1}(\boldsymbol{P}_t, \sigma)$ is introduced.

In L^n_{corr}, since numerator $\boldsymbol{P}_n$ and $\boldsymbol{P}_{n+1}$ follow the behavior of two independent $2W+1$ degree non-central normal distributions, the product of independent random variables become:

$$\begin{aligned} \boldsymbol{P}_n \cdot \boldsymbol{P}_{n+1} &\sim (\boldsymbol{P}_t + \mathscr{N}_{2W+1}(0,\sigma)) \cdot (\boldsymbol{P}_{t+1} + \mathscr{N}_{2W+1}(0,\sigma)) && (8) \\ &\sim \boldsymbol{P}_t \cdot \boldsymbol{P}_{t+1} + (\boldsymbol{P}_t + \boldsymbol{P}_{t+1}) \cdot \mathscr{N}_{2W+1}(0,\sigma) \\ &\quad + \mathscr{N}_{2W+1}(0,\sigma) \cdot \mathscr{N}_{2W+1}(0,\sigma) && (9) \end{aligned}$$

If both Gaussian random variables X and Y have zero mean and are independent, $Z = XY$ has the probability density of Bessel function:

$$\begin{aligned} p_Z(z) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \mathrm{d}x \int_{-\infty}^{\infty} \mathrm{d}y \, \mathrm{e}^{-(x^2+y^2)/2\sigma^2} \delta(z - xy) && (10) \\ &= \frac{1}{\pi\sigma^2} \mathrm{K}_0\left(\frac{|z|}{\sigma^2}\right) && (11) \end{aligned}$$

Therefore, Equation (8) can be rewritten as:

$$\begin{aligned} \boldsymbol{P}_n \cdot \boldsymbol{P}_{n+1} &\sim \boldsymbol{P}_t \cdot \boldsymbol{P}_{t+1} + (\boldsymbol{P}_t + \boldsymbol{P}_{t+1}) \cdot \mathscr{N}_{2W+1}(0,\sigma) \\ &\quad + \frac{1}{\pi} \mathrm{K}_0(|z|) && (12) \end{aligned}$$

We now take a close look at the denominator $\|\boldsymbol{P}_n\|$:

$$\begin{aligned} \|\boldsymbol{P}_n\| &= (P^2_{1,n} + P^2_{2,n} + \ldots + P^2_{2W+1,n})^{1/2} && (13) \\ &= \sigma \left(\sum_{i=1}^{2W+1} (\frac{P_{i,n}}{\sigma})^2 \right)^{1/2} \sim \sigma \chi^2(2W+1; \boldsymbol{P}_t/\sigma^2)^{1/2} && (14) \end{aligned}$$

$\|\boldsymbol{P}_n\|$ follows the behavior of non-central Chi-square distribution with $2W+1$ degrees of freedom. Therefore, cosine similarity of two vectors corrupted with Gaussian noise (Equation (7)) is:

$$\begin{aligned} &\frac{\boldsymbol{P}_n \cdot \boldsymbol{P}_{n+1}}{\|\boldsymbol{P}_n\| \|\boldsymbol{P}_{n+1}\|} \sim \\ &\frac{\boldsymbol{P}_t \cdot \boldsymbol{P}_{t+1} + (\boldsymbol{P}_t + \boldsymbol{P}_{t+1}) \cdot \mathscr{N}_{2W+1}(0,\sigma) + \frac{1}{\pi} \mathrm{K}_0(|z|)}{\sigma^2 \chi^2(2W+1; \boldsymbol{P}_t/\sigma^2)^{1/2} \chi^2(2W+1; \boldsymbol{P}_{t+1}/\sigma^2)^{1/2}} \end{aligned} \tag{15}$$

The probability density function remains complicated with the ratio distribution of Bessel distribution and product of two non central chi-square distributions. Therefore, we use simulation to study the performance of Equation (4) with regard to $\boldsymbol{\beta}$, together with its high-SNR counterpart $\frac{\boldsymbol{P}_t \cdot \boldsymbol{P}_{t+1}}{\|\boldsymbol{P}_t\| \|\boldsymbol{P}_{t+1}\|}$.

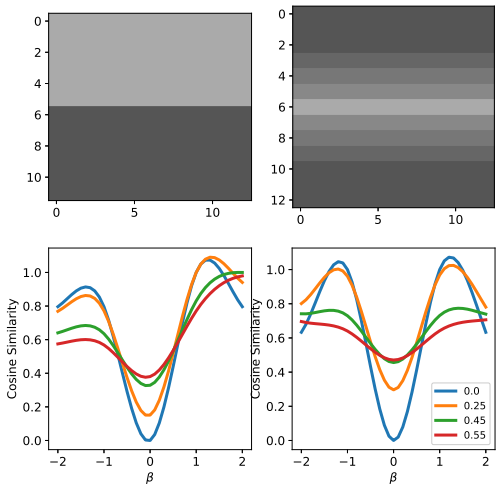

Figure 1. Objective function visualization with different SNR. Upper panels: simulated images without noise. Lower panels: corresponding objective functions in response to β with Gaussian noise added.

We simulate one edge (same as main Fig. 3) and one ridge. We set $N = 1$ for the convenience of visualization. We find that β_{min}, which corresponds to the minimum objective value, is invariant to different amplitudes of Gaussian noise (Figure 1). This simulation demonstrates that $\arg\min L^n_{\text{corr}}$ remains the same, meaning our method can be noise-tolerant up to $\sigma = 0.55$.

4. Computational complexity

Time complexity estimation. We here calculate the time complexity per iteration per unit length via multiple variable notation [7]. First, the time complexity of $Y(\boldsymbol{\beta}^{(e)}) = \sum_{n=0}^{N} \beta_n^{(e)} t^n$ is $\mathcal{O}(N)$, where N is degree of a polynomial defined in Main Sec. 3.1. Therefore, the total time cost of polynomial operation (step 6 in Algorithm 1) is $\mathcal{O}(2N)$. Next, 1D convolution operator $\boldsymbol{P}_t(\lfloor Y(\boldsymbol{\beta}^{(e)})\rfloor, \lfloor X(\boldsymbol{\alpha}^{(e)}\rfloor)) * \boldsymbol{N}_\Omega$ has a time cost of $\mathcal{O}((2W+1)^2)$, where W is the half of scanning window size, defined in Main Sec. 3.2. After convolution, cosine similarity is also calculated. Given that each of $\boldsymbol{P}_t$ and $\boldsymbol{P}_{t+1}$ has the size of $(2W+1)$. Time complexity of cosine similarity $\frac{\boldsymbol{P}_n \cdot \boldsymbol{P}_{n+1}}{\|\boldsymbol{P}_n\| \|\boldsymbol{P}_{n+1}\|}$ is $\mathcal{O}((2W+1)^2)$. Therefore the total time cost of step 7 is $\mathcal{O}((2W+1)^4)$. Both $L^{(e)}_{\text{reg}}$ and L_{cont} have a constant time complexity $\mathcal{O}(N)$. Hence the total time complexity before gradient calculation is $\sim \mathcal{O}((2W+1)^4)$ (Step 6-10)

Based on the auto-differentiation [2], Time complexity of gradient calculation is proportional to the number of parameters, which is $\mathcal{O}(2N(2W+1)^2)$. Next, parameters are updated at $\mathcal{O}(2)$. Hence the time cost from step 11-13 is $\sim \mathcal{O}(2N(2W+1)^2)$.

In summary, the total time cost of our method is $\sim \mathcal{O}\left(EL(2W+1)^4\right)$, where L is the integrated length of the curve and E is the total iterations before converge. Given that N and W are relatively small in value (Tab. 1), E and L dominate the time complexity of our method.

Space complexity estimation. Auto-differentiation [2] requires for all operations to build a graph. The space complexity of $Y(\boldsymbol{\beta})$ and $X(\boldsymbol{\alpha})$ is $\mathcal{O}(2N+2)$. $\boldsymbol{P}_t(\lfloor Y(\boldsymbol{\beta}^{(e)})\rfloor, \lfloor X(\boldsymbol{\alpha}^{(e)}\rfloor))$ has the space cost of $\mathcal{O}((2W+1)^2)$. Objective function has a constant complexity of $\mathcal{O}(3)$. Therefore, $\sim \mathcal{O}((2W+1)^2)$ per iteration is required to perform automatic differentiation. At last, $\mathcal{O}(3NK)$ is required to describe the curves sufficiently after parametric tracing is complete, where K is the number of knots defined in Main Sec. 3.1.

It is worth mentioning that we here only provide the estimated time complexity and space complexity as they also depend on the choice of gradient-based algorithms.

5. Segmented results of lane following

Shown in Figure 2

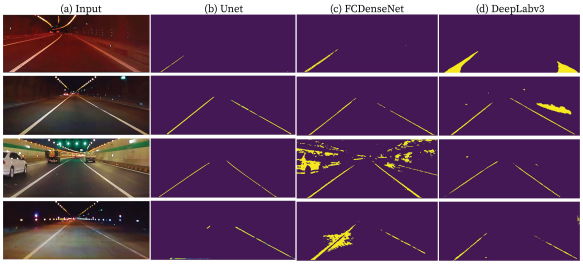

Figure 2. A few examples of segmented results using DCN under various lighting conditions, including bright and dim lighting: (a) Inputs, (b) U-Net [12], (c) FCDenseNet [8,9], (d) DeepLabv3+ [4] + MobileNetv2 [13].

6. Numerical Experiment on Seismic dataset

We here continue to evaluate the performance of our method with seismic data, a spacial application. In contract to applications shown in main text, where curves are following any arbitrary non-smooth (SPT/SMT) and smooth (lane following) response, the underlying parametric model to describe a curve in seismic data is often known to be polynomial.

Seismic introduction. A seismic gather is a collection of seismic traces, where a detector source sends a compressional wave through a medium (in our case, layers of rock) and a receiver records the amplitudes reflected back. These reflections are caused by a refractive index mismatch between rock layers, where there is an impedance change between the density of the layers.

Noise level	$\sigma = 0$			$\sigma = 0.1$			$\sigma = 0.3$			$\sigma = 0.55$		
Methods	$\mathscr{L}$	$\mathscr{D}$	$\mathscr{V}$	$\mathscr{L}$	$\mathscr{D}$	$\mathscr{V}$	$\mathscr{L}$	$\mathscr{D}$	$\mathscr{V}$	$\mathscr{L}$	$\mathscr{D}$	$\mathscr{V}$
NLM [3] + Otsu's [10]	-	-	-	0.882	6.651	0.836	0.871	16.038	2.103	0.881	14.825	1.957
BM3D [5] + Otsu's [10]	-	-	-	0.866	17.870	2.353	0.841	17.912	2.402	0.819	17.551	2.351
FMM [11] $\epsilon = 0.01$	-	2.015	0.793	-	2.252	0.344	-	2.010	0.324	-	1.850	0.304
FMM [11] $\epsilon = 1$	-	4.190	0.653	-	2.314	0.352	-	2.083	0.331	-	1.916	0.311
UNet [12]	0.691	1.136	0.276	0.697	1.641	0.399	0.721	1.802	0.419	0.732	1.972	0.456
FC-DenseNet [9]	**0.933**	1.560	0.308	**0.906**	1.465	0.307	0.880	2.242	0.476	0.850	1.601	0.362
BDCU-Net [1]	0.904	0.834	0.169	0.895	0.989	0.205	0.883	0.904	0.195	0.877	1.086	0.250
Ours	0.931	**0.529**	**0.124**	0.892	**0.529**	**0.121**	**0.889**	**0.616**	**0.138**	**0.882**	**0.658**	**0.193**

Table 5. Evaluation results of seismic event picking. Best in bold.

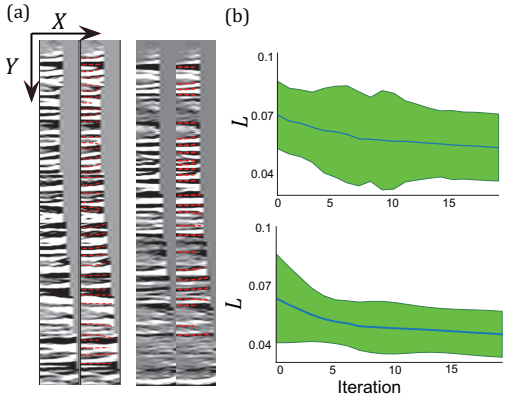

Figure 3. Objective function value decrease after a few iterations during parametric curves tracing in seismic gathers. (a) Two raw seismic gathers and their parametric curves overlaid on top of the inputs (Best viewed with zoom). (b) Average L of the corresponding gather during iterative optimization.

Since a reflection at a given depth (Figure 3, y-axis) can be observed at different reflection angles between source and receiver, these traces are "gathered" by the angle (Figure 3, x-axis) to form an image. In many seismic imaging workflows, collecting a parametric curve along a reflecting event is vitally important because curved events represent errors in the estimate of the velocity of the subsurface layers, as the arrival times of signal differ by the distance traveled in the medium. Horizontally "flat" events indicate that the velocity model of the subsurface is correct.

Traditionally, this information is collected by fitting the second-order polynomials: $y = a_0 + a_1 x + a_2 x^2$, $x \in [0, X]$. This means that the underlying model of the curves is known and there's only one time interval ($K = 1$). In contrast, the underlying parametric model for the examples shown in the main text is unknown. Therefore, we construct a dataset containing around 1000 seismic gathers with a broad range of depth (Figure 3, y-axis) and scattered locations, reflecting various seismic conditions in every spatial dimension. Ground truth is annotated manually by experts. To improve the accuracy, annotations with the highest certainty are selected, leading to sparse labeling.Different amplitudes of Gaussian noise ($\sigma = 0.0, 0.1, 0.3$ and 0.55) are added to test the robustness of our method under low SNR.

Results. Figure 3 illustrates two examples in the testing set, together with the curve tracing results and objective function value decrease during optimization. Using metrics defined in the main text, we also show the results and visual comparison among all competing methods in Tab. 5 and Figure 4. However, due to the sparse annotation, we only compare the output curves which overlap with the ones labeled in the ground truth by calculating the distance of 0-th coefficient between outputs and ground truth (GT): $\|a_{0,GT} - a_{0,Output}\| < 1$.

Unlike the results from SPT/SMT where all methods perform well in high SNR, the high-density and possibly overlapping signals presented in the seismic dataset pose a challenge for traditional methods and FMM to trace the curves accurately even in high SNR. Among all DCN methods, BDCU-Net has better accuracy in terms of $\mathscr{D}$ and $\mathscr{V}$ while FCDenseNet is detecting longer curves $\mathscr{L}$. However, our method outperforms the state-of-the-art methods in $\mathscr{D}$ and $\mathscr{V}$ significantly since we directly optimize polynomial parameters. Our approach can also detect curves efficiently, comparable to the best performance of CDN methods.

References

[1] Reza Azad, Maryam Asadi, Mahmood Fathy, and Sergio Escalera. Bi-directional convlstm u-net with densley connected convolutions. pages 406–415, 10 2019. 4

[2] Atılım Günes Baydin, Barak A. Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: A survey. *J. Mach. Learn. Res.*, 18(1):5595–5637, Jan. 2017. 3

[3] A. Buades, B. Coll, and J. . Morel. A non-local algorithm for image denoising. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 60–65 vol. 2, 2005. 4

[4] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, 2018. 3

[5] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-d transform-domain collaborative fil-

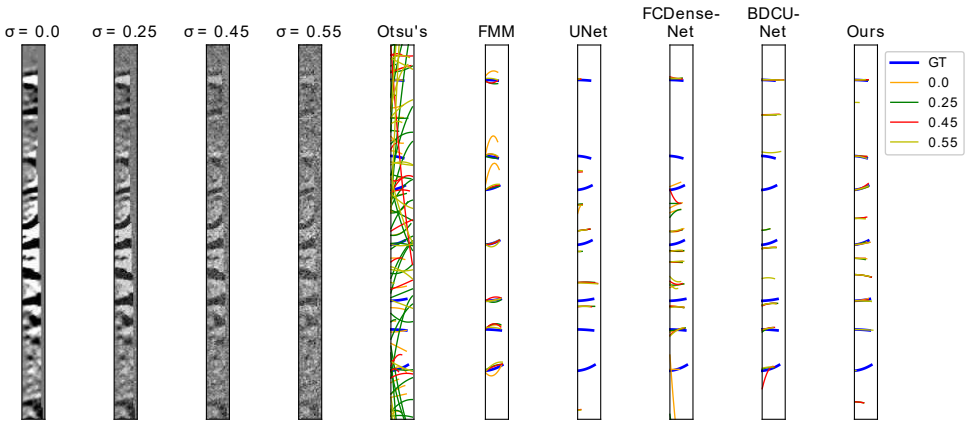

Figure 4. Visual comparison of seismic dataset. Note that images are normalized to 1.

tering. *IEEE Transactions on Image Processing*, 16(8):2080–2095, 2007. 4

[6] David L Donoho and Iain M Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, 09 1994. 1, 2

[7] R. Howell. On asymptotic notation with multiple variables. 2008. 3

[8] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, 2017. 3

[9] S. Jégou, M. Drozdzal, D. Vazquez, A. Romero, and Y. Bengio. The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1175–1183, 2017. 3, 4

[10] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, 1979. 4

[11] Gabriel Peyré, Mickaël Péchaud, Renaud Keriven, and Laurent D. Cohen. Geodesic Methods in Computer Vision and Graphics. *Foundations and Trends in Computer Graphics and Vision*, 5(3-4):197–397, 2010. 4

[12] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing. 3, 4

[13] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *CVPR*, 2018. 3