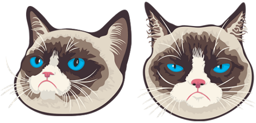
Speeding up wave propagation modeling
CheckPoint # 2

Team: GrumpyCat
Ivan Gerasimov, Olha Tsymboi

Moscow Institute of Physics and Technology

November 15, 2020



Introduction & Problem statement

Aim:

- To speed up acoustic equation solution using using neural networks.

Input:

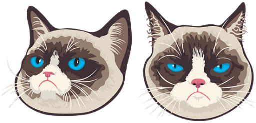
- time and space discretizations Δt, $\Delta x = \Delta z$
- time and space pints amout n_t, n_x, n_z
- impulse source time-series $q(t_i)$ and location x_s, z_s.
- special velocities $vp(x,z)$ at data points
- absorbing boundary conditions

Output:

- The solution of acoustic equation $u(x,z,t)$ at some point of time t.

Quality:

- RMSE error between normalized wavefields.



Data generation: velocity modeling

- Geo-realistic Marmousi-II model [1].
- Texture transferring using Random Gaussian field context [1].

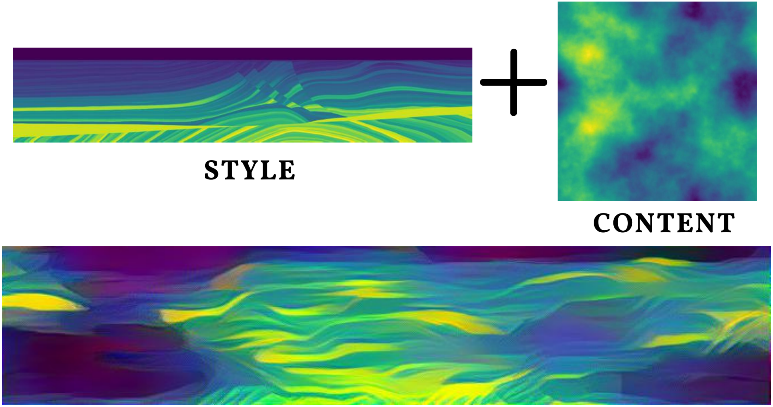


Figure: NST - style: Marmousi model, content: Gaussian field

Data generation: velocity modeling

- Image size adjustment and Gaussian smoothing.

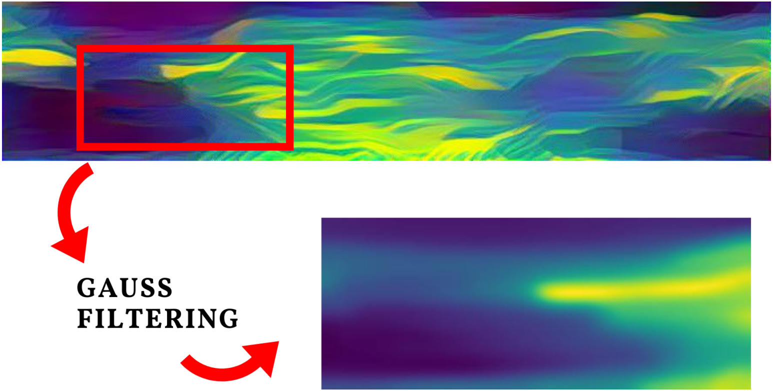


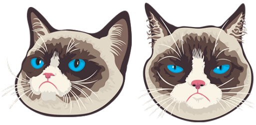
Figure: Crop and apply Gauss Filtering

Data generation: velocity modeling

- Geo-realistic scaling.

$$\left.\begin{array}{r} 1)\ \mathrm{vp} = \mathrm{vp} - \mathrm{vp.min()} \\ 2)\ \mathrm{vp} = \dfrac{\mathrm{vp}}{\mathrm{vp.max()}} \end{array}\right\} \text{transform to scale } [0, 1]$$

$$\left.\begin{array}{r} [\mathrm{low},\ \mathrm{high}] \in [500, 10000] \\ 3)\ \mathrm{vp} = \mathrm{vp} * (\mathrm{high} - \mathrm{low}) + \mathrm{low} \end{array}\right\} \text{transform to scale } [\mathrm{low},\ \mathrm{high}]$$



Data generation

Generation:

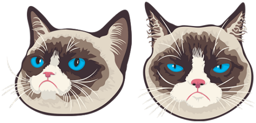
- Velocity models generation $vp(x,z)$:
- Nyquist frequency varying N_λ
- Space discretization varying $\Delta d,\ \Delta z = \Delta x = \Delta d$
- Ricker source with varying frequency f_0 under the condition

$$\begin{cases} \Delta d \cdot f0 & \in [1,\ 256] \\ \Delta d \cdot f0 & \leq \frac{\text{vp.min()}}{N_\lambda} \end{cases}$$

- Boundary conditions with varying a:
- Discretizations to fulfill CFL condition:

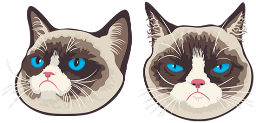
$$\Delta t \leq \frac{\Delta x}{\sqrt{2}|vp(x,z)|.\max()}$$

- Random source locations



Model:

- Convolutional Auto-Encoder + L1Loss/MSE
- Convolutional Auto-Encoder + GRU Cell at the bottleneck + L1Loss/MSE
- UNet (UNet++)[6] + L1Loss/MSE
- Fully-Connected + Physic's Informed Loss [3]
- Convolutional + Physic's Informed Loss



Model:

- Convolutional Auto-Encoder + L1Loss/MSE

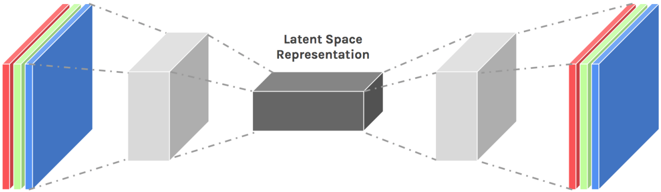
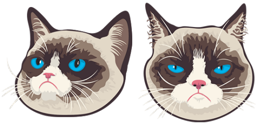


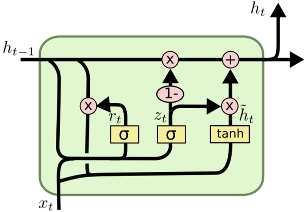
Figure: Convolutional Auto-Encoder

Input image: $u(x,z,t_i) + \frac{q(t_i)\Delta x^2}{\Delta t^2} \cdot \delta(x - x_s, z - z_s),\ vp(x,z)$



Model:

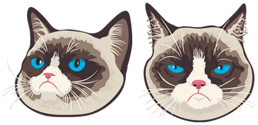
- Convolutional Auto-Encoder + GRU Cell at the bottleneck + L1Loss/MSE



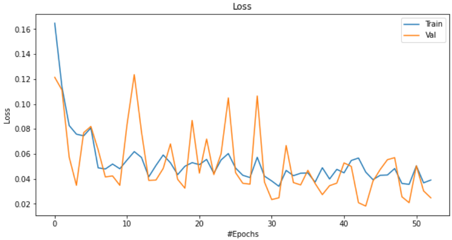
$$
\begin{aligned}
z_t &= \sigma(W_{hz} * h_{t-1} + W_{xz} * x_t + b_z) \\
r_t &= \sigma(W_{hr} * h_{t-1} + W_{xr} * x_t + b_r) \\
\hat{h}_t &= \Phi(W_h * (r_t \odot h_{t-1}) + W_x * x_t + b) \\
h_t &= (1 - z_t) \odot h_{t-1} + z \odot \hat{h}_t
\end{aligned}
$$

Figure: Convolutional GRU Cell [5]

Input image: $u(x, z, t_i) + \frac{q(t_i)\Delta x^2}{\Delta t^2} \cdot \delta(x - x_s, z - z_s)$
Hidden initial image: $vp(x, z)$



Results: CNN-AE + L1Loss



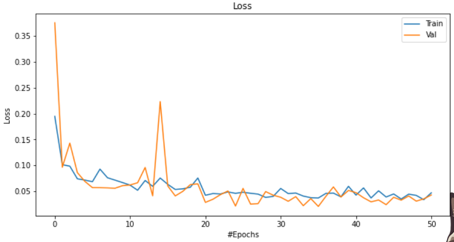
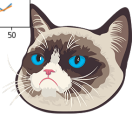
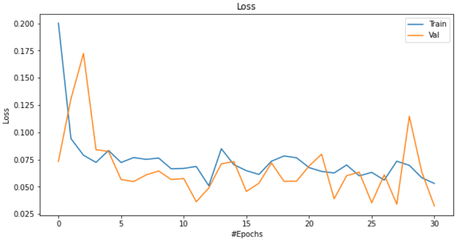


Figure: ELU, ReLU activations



Results: CNN-GRU-AE + L1Loss



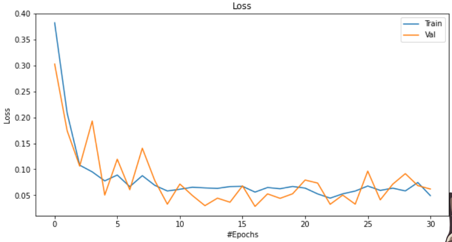
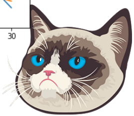
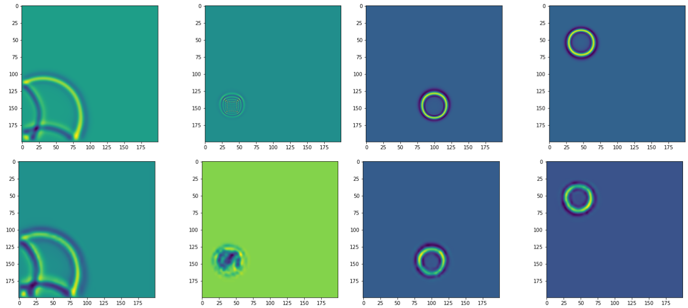


Figure: ELU, ReLU activations



Discussion



- Try deeper networks
- Random area sub-sampling near source location
- Random time sub-sampling for wider time-horizons
- Add skip-connections
- Realize FC-PIL, Conv-PIL, GAN based predictions.
- Provide exact time performance

References I

- D. Bevc and O. Nedorub.
  SEG Technical Program Expanded Abstracts 2019.
  Society of Exploration Geophysicists, 2019.
- M. Louboutin, M. Lange, F. Luporini, N. Kukreja, P. A. Witte, F. J. Herrmann, P. Velesko, and G. J. Gorman.
  Devito (v3.1.0): an embedded domain-specific language for finite differences and geophysical exploration.
  Geoscientific Model Development, 12(3):1165–1187, 2019.
- B. Moseley, A. Markham, and T. Nissen-Meyer.
  Solving the wave equation with physics-informed deep learning, 06 2020.
- A. Siahkoohi, M. Louboutin, and F. Herrmann.
  Neural network augmented wave-equation simulation, 09 2019.
- M. Siam, S. Valipour, M. Jagersand, and N. Ray.
  Convolutional gated recurrent networks for video segmentation, 2016.

References II

Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang.
Unet++: Redesigning skip connections to exploit multiscale features in image segmentation, 2020.

Thank you!

Questions?