# Splash in a Flash:
# Sharpness-aware minimization for efficient liquid splash simulation


**V. Jetly , H. Ibayashi and A. Nakano**

**Department of Computer Science, University of Southern California**


## OVERVIEW

- Machine learning based fluid simulations provide significant performance improvement [1, 2] by learning the underlying physics of droplet formation.
- However, computationally intensive training of these machine learning models on large three-dimensional datasets is a major drawback.
- To address this, we propose sharpness-aware minimization for machine learning based FLIP, which can efficiently learn the plausible dynamics of liquid splashes due to its ability to achieve robust solutions.
- Our results indicate the optimizations with SAM to rapidly converge and reduce training times.

## RELATED WORK

- **Data driven fluid dynamics:** Data driven approach has been popular in fluid simulation to eschew the heavy computation of Navier-Stokes equation. [1] pioneered this line of research and more recently neural networks has been adopted to improve various aspects of fluid simulation, such as projection solver [5], visual enhancement by image processing [6] and our target liquid splash simulation [3]
- **Sharpness of neural network:** The common training algorithms, such as SGD and Adam, only use first order gradient of loss function while the importance of second order gradient, sharpness, has been pointed out. SAM is the first training algorithm that can efficiently minimize sharpness [4].

## SAM OVERVIEW

- Sharpness-aware minimization (SAM) is a training scheme which minimizes the curvature of loss surface (sharpness) along with the training loss. [3]
- For training data $(\mathbf{x}, \mathbf{y})$ and training loss $L_w(\mathbf{x}, \mathbf{y})$, SAM defines sharpness as:

$$\max_{\|\epsilon\|_2 \le \rho} \{L_{w+\epsilon}(\mathbf{x}, \mathbf{y}) - L_w(\mathbf{x}, \mathbf{y})\}$$

- SAM minimizes this sharpness along with the training loss $L_w(\mathbf{x}, \mathbf{y})$ by the following update rule:

$$w = w - \eta \left( \nabla_w L_w(\mathbf{x}, \mathbf{y}) + \nabla_{w'} L_{w'}(\mathbf{x}, \mathbf{y})|_{w'=w+\rho} \frac{\nabla_w L_w(\mathbf{x}, \mathbf{y})}{\|\nabla_w L_w(\mathbf{x}, \mathbf{y})\|} \right)$$

## MLFLIP with SAM

- **Data Generation:**
  - The training data are generated through multiple high resolution FLIP simulations.
  - From these simulations, we extract feature vector **x**
    $\mathbf{x} \leftarrow$ {27×3 velocity values} + {27×1 level set values}
  - We use $10^6$ training examples from 16 different FLIP simulations using a grid spacing of 5 mm, initialized with random values, with even distribution of both splashing and non-splashing particles.

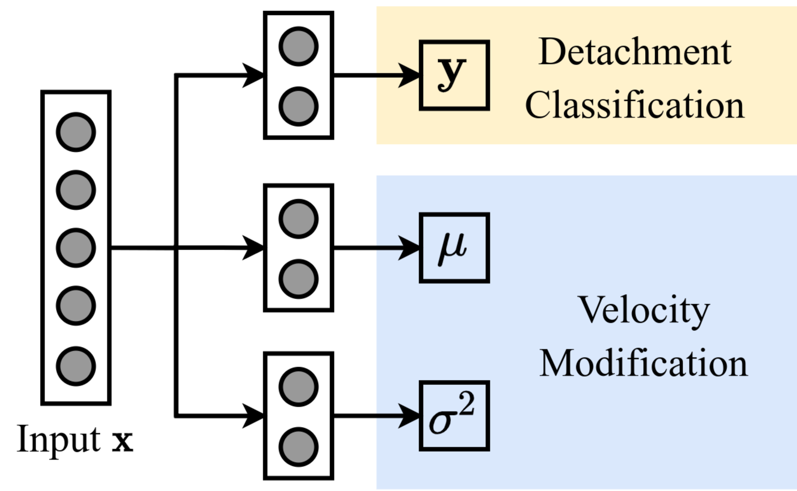


**Figure 1.** *Neural network layout used in this study.*

- **Neural Network Architecture:**
  - Inputs: Feature vector **x, (**Dimension: 108×1)
  - Outputs: Neural network predicts 2 components
    1. Detachment Classification: Binary classification of whether the input region is splash or not.
    2. Velocity Modification: Determines mean and variance of velocity change for a splash with respect to the fluid motion
  - Hidden layer:
    - ▪ 64 neurons
    - ▪ 10% dropout
    - ▪ *tanh* nonlinear activation
    - ▪ Batch Normalization

- **Loss Functions:**

1. For Detachment classifier:

$$L_d(\hat{\mathbf{y}}|\mathbf{x}) = -\sum_{i=1}^{n} \log P(\hat{y}_i|x_i)$$

2. For velocity modification:

$$L_v(\Delta\mathbf{v}|\mathbf{x}) = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{d}\left[\frac{(\Delta\mathbf{v}_{i,j} - \mu_{i,j})^2}{\sigma_{i,j}^2} + \ln\sigma_{i,j}^2\right]$$

Overall loss function, $L_d + L_v$ is updated using SAM

## RESULTS

- We trained two instances of the neural network:
  1. MLFLIP, reproduced from [3]
  2. MLFLIP + SAM
- Both models iteratively learn to capture realistic behavior of the fluids and gradually converge to faithfully represent the underlying physics of droplet formation.
- Optimization with SAM helps the model to rapidly converge, achieving a speedup of 3.76x over MLFLIP
- These promising results indicate the potential of SAM for fluid simulation and future works should explore the use of SAM for broader application of fluid dynamics such as pressure solvers and generative fluid models.

| | Epochs to converge | Wall-clock time |
|---|---|---|
| MLFLIP | 320 epochs | 2144 sec |
| **MLFLIP + SAM** | **50** epochs | **569** sec |

**Table 1.** *Performance comparison of ML based FLIP (MLFLIP)[3] without and with SAM optimization. Our training scheme drastically improves convergences speed both in number of epochs and wall-clock time while it simulates plausible water splash effect (Fig. 2 bottom)*

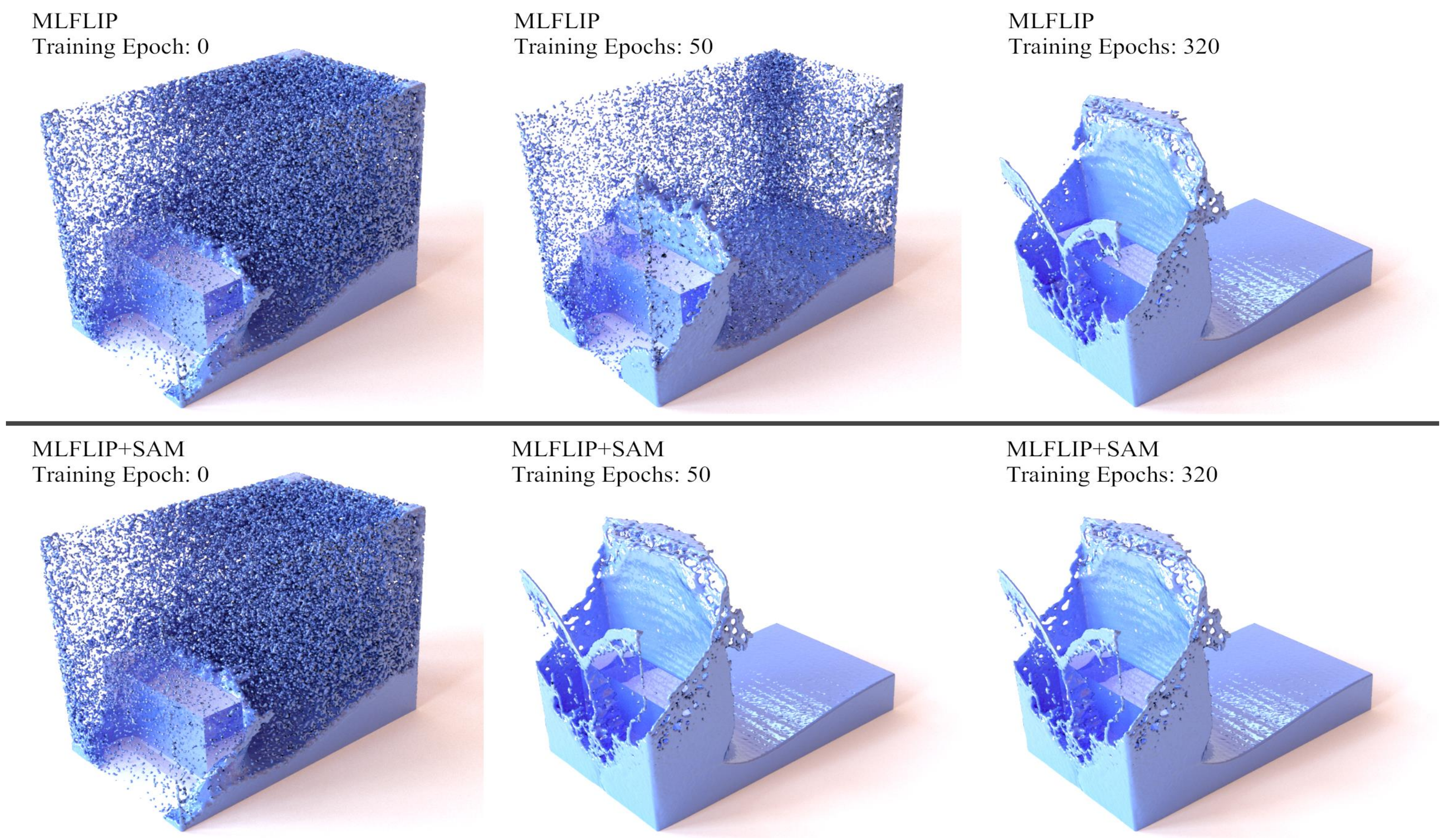


**Figure 2.** *Simulation results with the same number of training epochs. Our training scheme with sharpness-aware minimization quickly learns physical fluid dynamics (bottom) while the baseline result (top) with naive training scheme shows numerous unphysical water splashes.*

## REFERENCES


[1] Ladický, L. U., Jeong, S., Solenthaler, B., Pollefeys, M., & Gross, M. (2015). Data-driven fluid simulations using regression forests. *ACM Transactions on Graphics (TOG), 34(6)*, 1-9.

[2] Umetani, N., & Bickel, B. (2018). Learning three-dimensional flow for interactive aerodynamic design. *ACM Transactions on Graphics (TOG), 37(4)*, 1-10.

[3] Um, K., Hu, X., & Thuerey, N. (2018, December). Liquid splash modeling with neural networks. In *Computer Graphics Forum* (Vol. 37, No. 8, pp. 171-182).

[4] Foret, P., Kleiner, A., Mobahi, H., & Neyshabur, B. (2020). Sharpness-aware minimization for efficiently improving generalization.*sarXiv preprint arXiv:2010.01412.*

[5] Tompson, Jonathan, et al. "Accelerating eulerian fluid simulation with convolutional networks." *International Conference on Machine Learning*. PMLR, 2017.

[6] Chu, Mengyu, and Nils Thuerey. "Data-driven synthesis of smoke flows with CNN-based feature descriptors." *ACM Transactions on Graphics (TOG)* 36.4 (2017): 1-14.


USC University of Southern California

EG'22 TheEvent