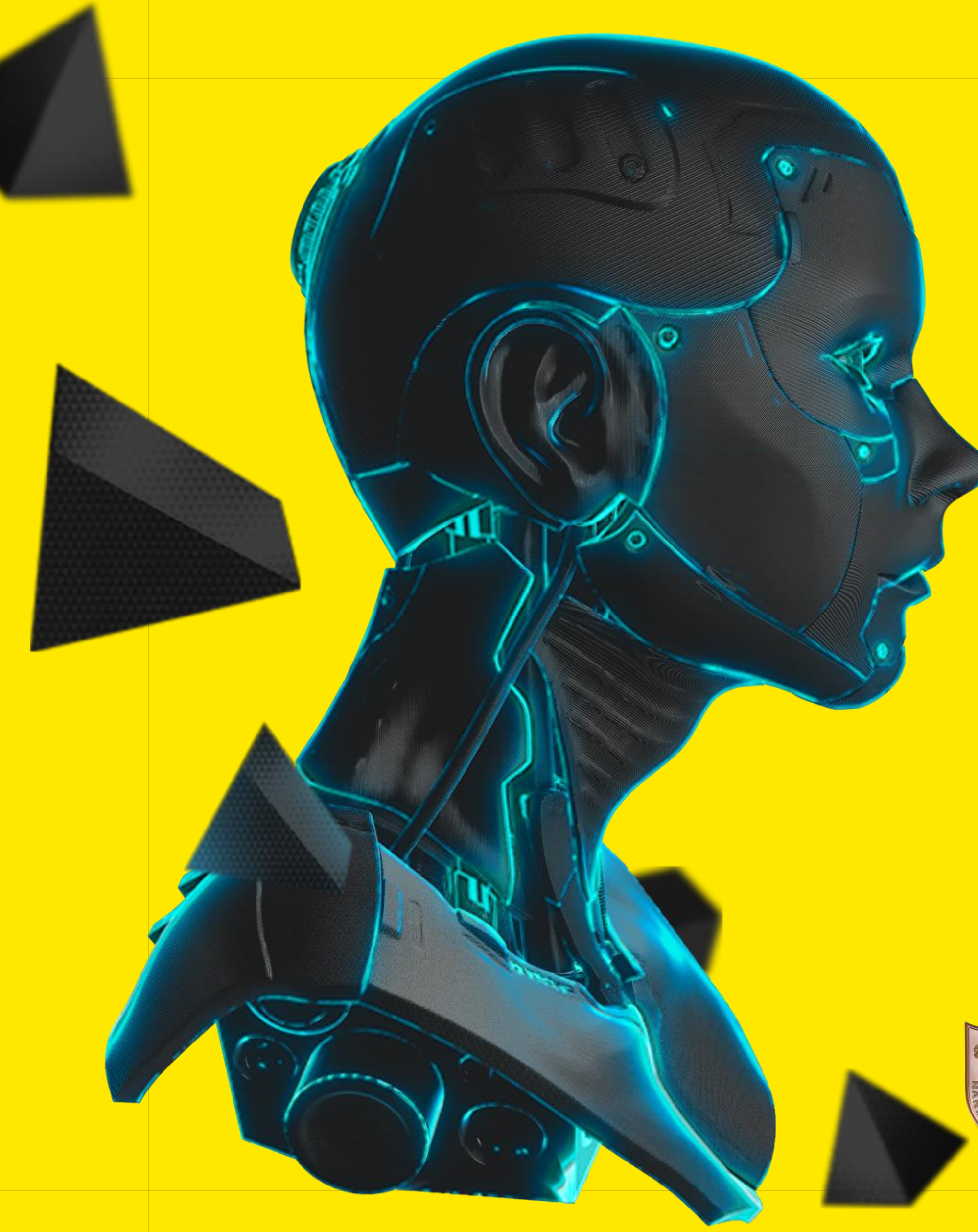


SIGGRAPH 2021

VOLUMETRIC APPEARANCE STYLIZATION WITH STYLIZING KERNEL PREDICTION NETWORK

Jie Guo [1] Mengtian Li [1,2] Zijing Zong [1] Yuntao Liu [1]

Jingwu He [1] Yanwen Guo [1] Ling-Qi Yan [3]

[1] State Key Lab for Novel Software Technology, Nanjing University

[2] Kuaishou Technology

[3] University of California, Santa Barbara

MOTIVATION

- Simulating the visual effects of heterogeneous media in CG requires a set of material properties:
  - Phase function — analytical model, e.g., the Henyey-Greenstein model
  - Density field — physically-based fluid simulations
  - Albedo field — relatively difficult to edit
- Our goal — artistically control the appearance of a medium according to only one guided image

RELATED WORKS

SIGGRAPH 2021

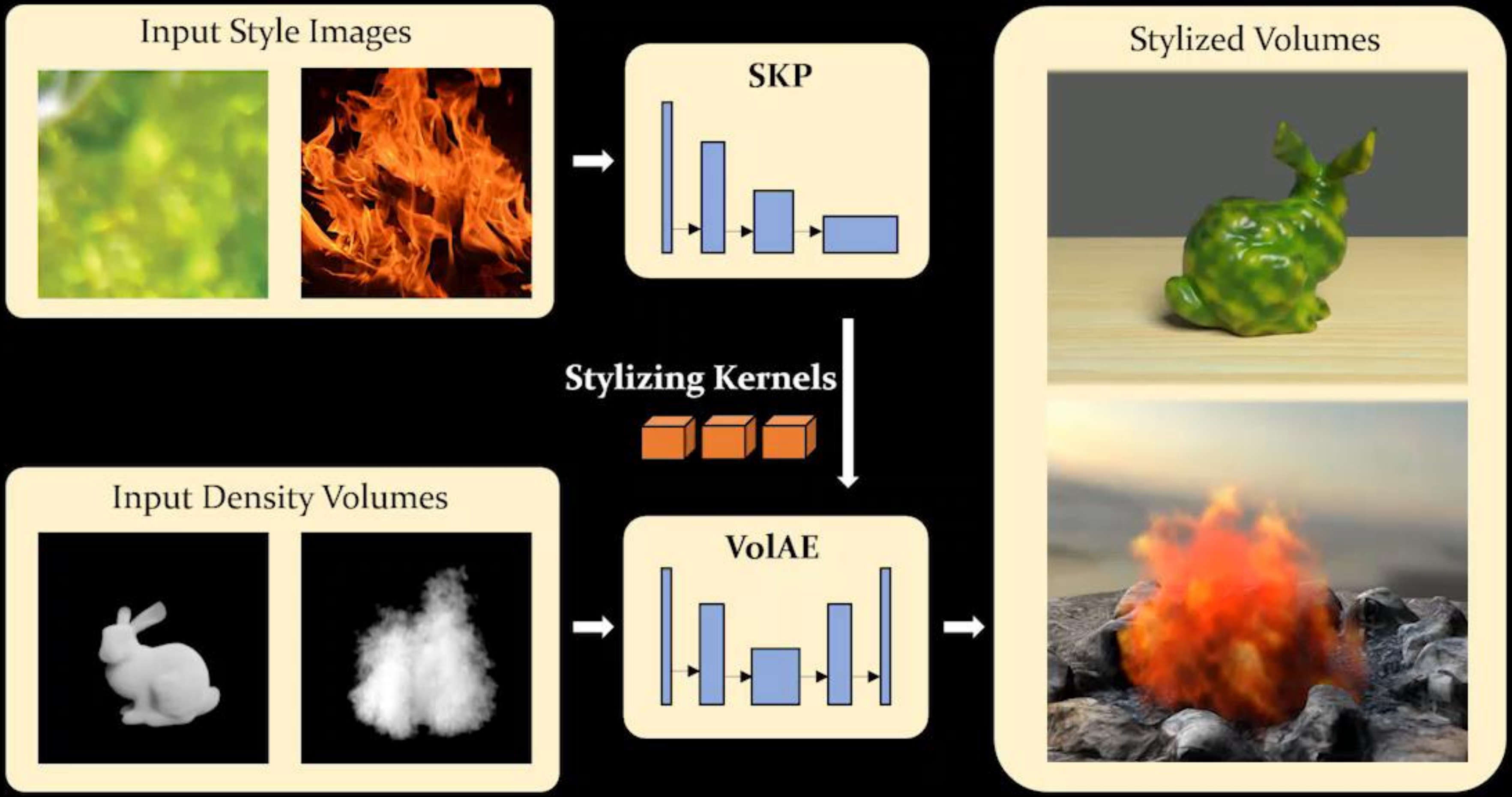
Method	Objective	Iterative Optimization	Performance
LazyFluid [Jamriška et al. 2015]	2D image	√	≈ 60s CPU
PhotoWCT [Li et al. 2018]	2D image	×	≈ 1s GPU
Tomography [Klehm et al. 2014]	3D density + albedo	√	≈ 1s GPU
TNST [Kim et al. 2019]	3D density	√	≥ 60s GPU
LNST [Kim et al. 2020]	3D density + 2D albedo	√	≈ 5s GPU

OUR METHOD

- ✓ Trained in unsupervised manner
- ✓ Support arbitrary style transfer
- ✓ Support both static and dynamic volumes
- ✓ Fast to evaluate (less than one second)

OVERVIEW

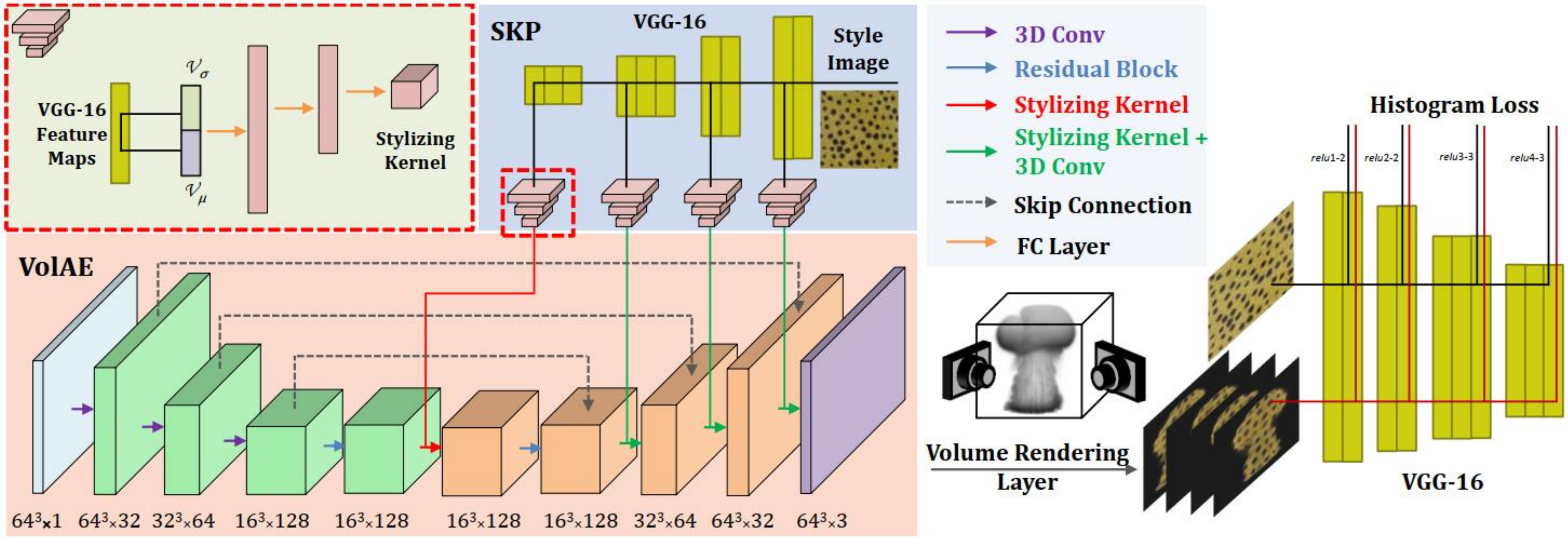
SIGGRAPH 2021



NETWORK ARCHITECTURE

SIGGRAPH 2021

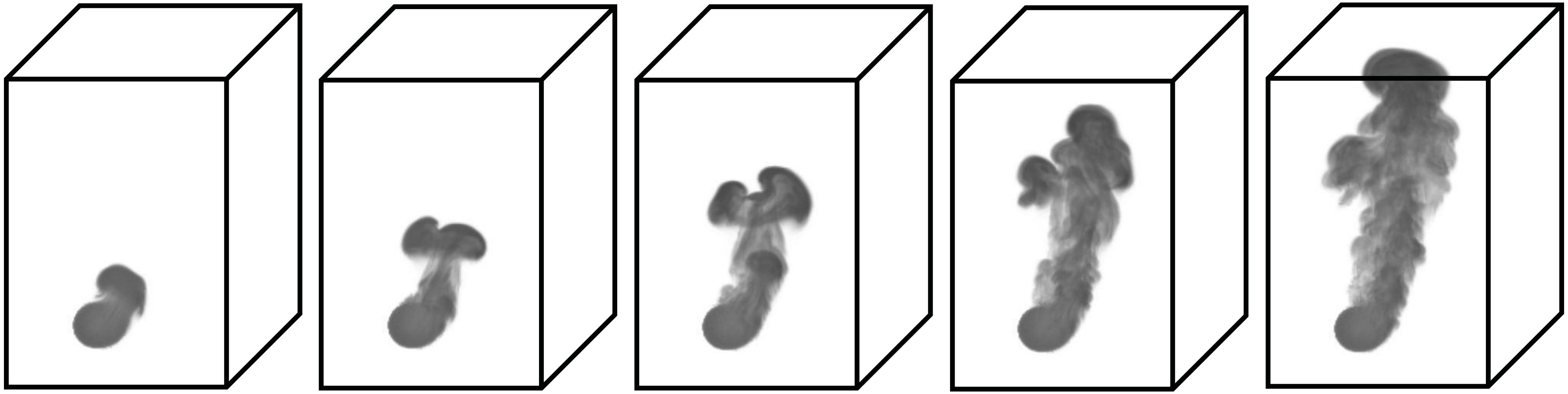
- Key components
  - Stylizing kernel & Its predictor
  - Volume autoencoder & Density-aware instance normalization
  - Volume rendering layer



DENSITY-AWARE INSTANCE NORMALIZATION

SIGGRAPH 2021

- The distribution of density usually varies considerably in the process of smoke simulation
- Influence the computation of the mean and variance used in standard Instance Normalization



$\mu = 0.0082$	$\mu = 0.0113$	$\mu = 0.0149$	$\mu = 0.0192$	$\mu = 0.0241$
$\sigma^2 = 0.0075$	$\sigma^2 = 0.0092$	$\sigma^2 = 0.0104$	$\sigma^2 = 0.0111$	$\sigma^2 = 0.0119$

DENSITY-AWARE INSTANCE NORMALIZATION

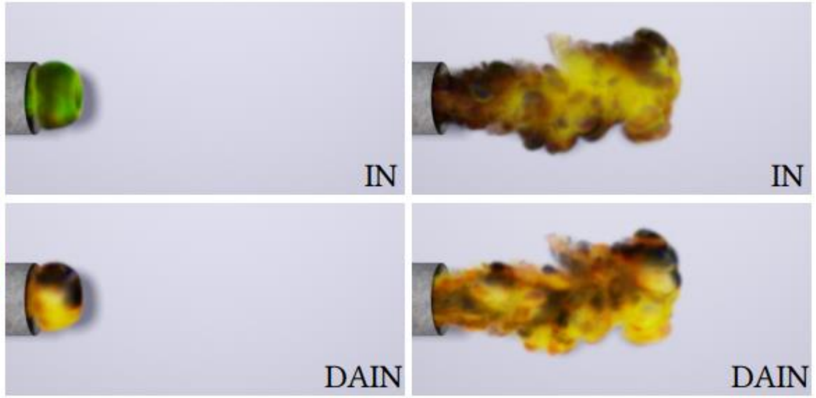
- Obtain a smoothing density mask

$$\mathbf{m} = 1 - \exp(-\lambda \mathbf{V}_\sigma^2)$$

- Normalize with weighted mean and variance

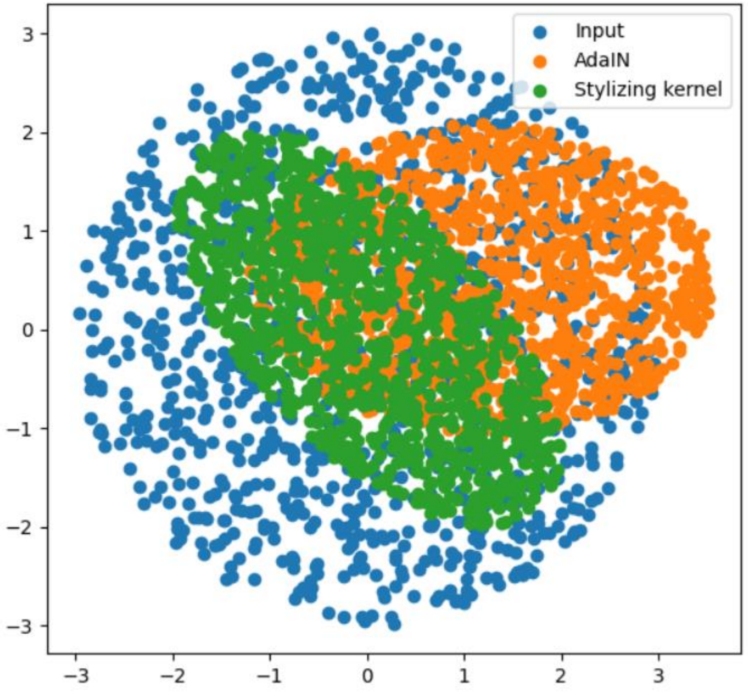
$$\mu_c = \frac{1}{\sum_{i,j,k} \mathbf{m}_{i,j,k}} \sum_i^D \sum_j^H \sum_k^W \mathcal{F}_{c,i,j,k} \cdot \mathbf{m}_{i,j,k}$$

$$\sigma_c^2 = \frac{1}{\sum_{i,j,k} \mathbf{m}_{i,j,k}} \sum_i^D \sum_j^H \sum_k^W (\mathcal{F}_{c,i,j,k} - \mu_c)^2 \cdot \mathbf{m}_{i,j,k}$$

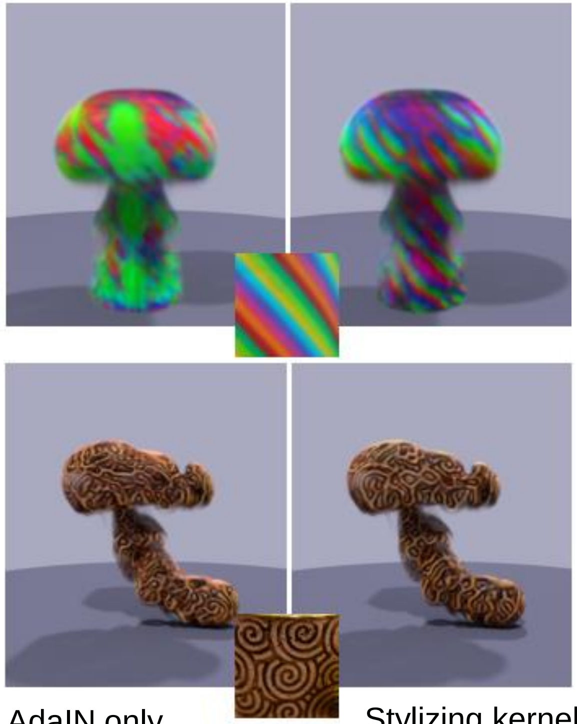


STYLIZING KERNEL

- Consist of an adaptive Conv3d and an adaptive DAIN
- Inject style information across different feature channels



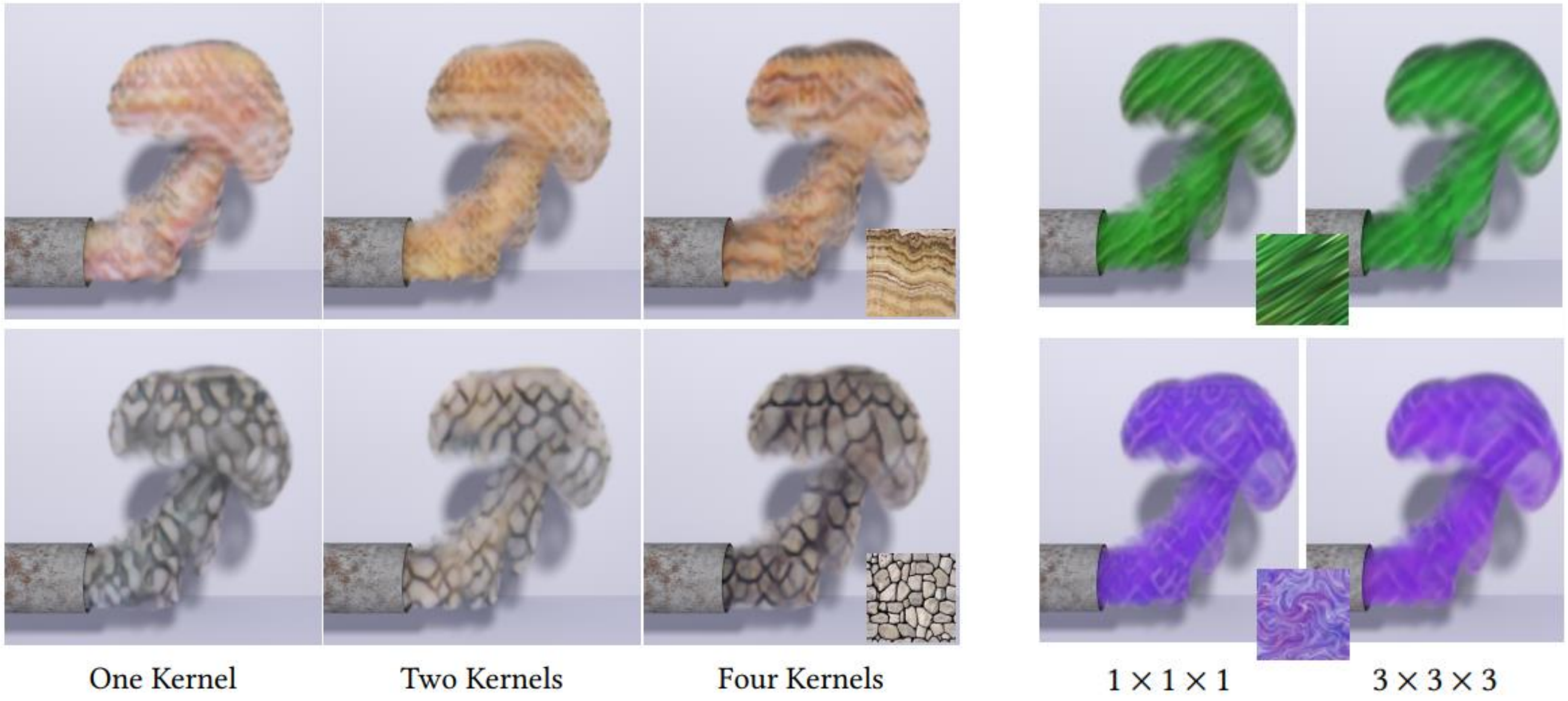
2D toy example



AdaIN only   Stylizing kernel

STYLIZING KERNEL

- Comparison on the number and size of stylizing kernel



VOLUME RENDERING LAYER

- The Radiative Transfer Equation

$$\frac{\mathrm{d}L_\mathrm{o}(p,\omega)}{\mathrm{d}t} = \underbrace{-\sigma_\mathrm{a} L_\mathrm{i}(p,\omega)}_{\text{Absorption}} \underbrace{-\sigma_\mathrm{s} L_\mathrm{i}(p,\omega)}_{\text{Out-scattering}} + \underbrace{L_\mathrm{e}(p,\omega)}_{\text{Emission}} + \underbrace{\sigma_\mathrm{s}\int_\Omega p(\omega_i \to \omega) L_\mathrm{i}(p,\omega_i)\,\mathrm{d}\omega_i}_{\text{In-scattering}}$$

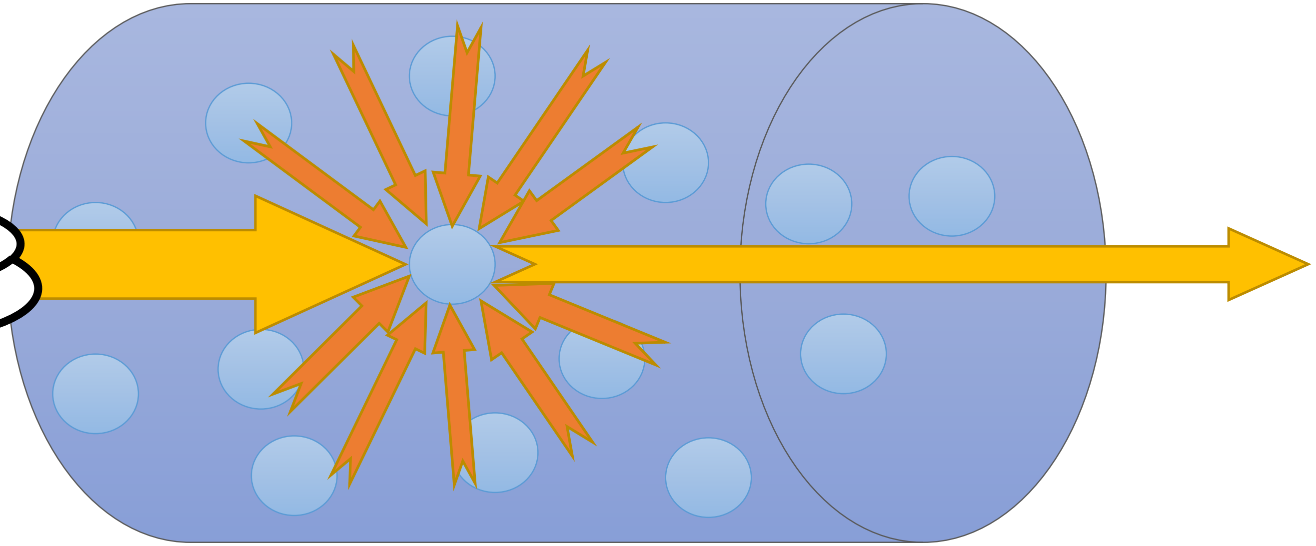
Extinction coefficient

$$\sigma_\mathrm{t} = \sigma_\mathrm{a} + \sigma_\mathrm{s}$$

Albedo

Our network learns a mapping from σ_t to α

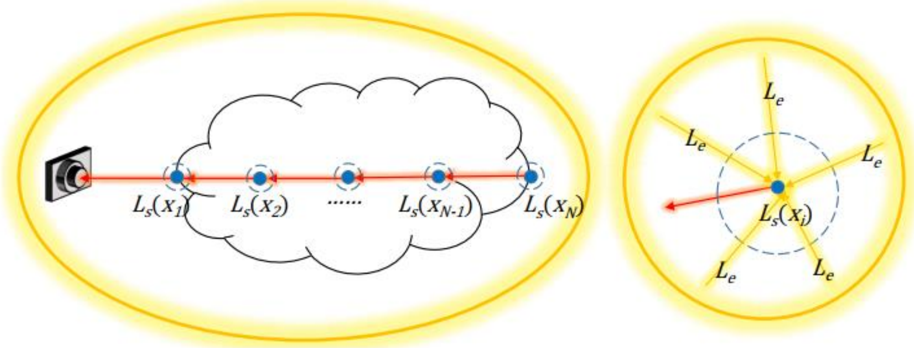
$$\sigma_\mathrm{s} = \alpha \cdot \sigma_\mathrm{t}$$



VOLUME RENDERING LAYER

SIGGRAPH 2021

- Assumptions
  - Isotropic phase function
  - Constant environmental light
  - Single scattering



- Under these assumptions, the volume rendering layer is to evaluate

$$L(\mathbf{x},\omega) = \int_0^t Tr(\mathbf{x}_t \to \mathbf{x})\sigma_t(\mathbf{x}_t)\alpha(\mathbf{x}_t)L_s(\mathbf{x}_t)\mathrm{d}t$$

$$Tr(\mathbf{x}_t \to \mathbf{x})$$

Precomputed and stored in a grid

- Forward render via ray marching

$$L(\mathbf{x},\omega) = \sum_{i=1}^{N} \exp\left\{-\sum_{j=1}^{i}\sigma_t(\mathbf{x}_j)\Delta t\right\}\sigma_t(\mathbf{x}_i)\alpha(\mathbf{x}_i)L_s(\mathbf{x}_i)\Delta t$$

$$L_s(\mathbf{x}_i) = \frac{L_e}{K}\sum_{k=1}^{K} Tr(\mathbf{x}_{b,k} \to \mathbf{x}_i)$$

VOLUME RENDERING LAYER

SIGGRAPH 2021

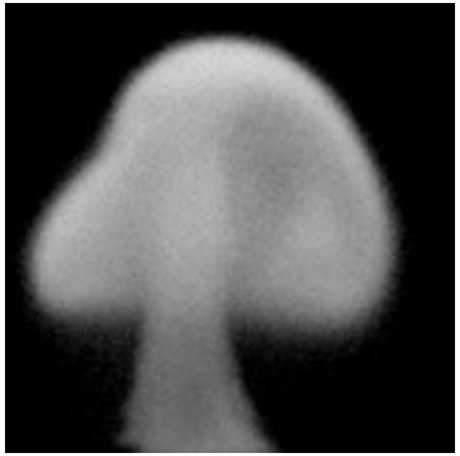
Backward render via ray marching, too

- Derivative of volume rendering layer

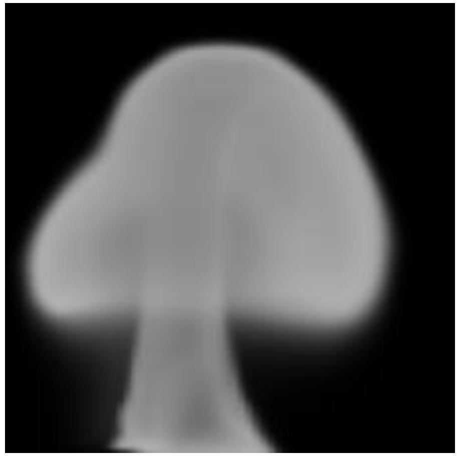
$$\frac{\partial L(\mathbf{x}, \omega)}{\partial \alpha(\mathbf{x}_i)} = \exp\left\{-\sum_{j=1}^{i} \sigma_{\mathrm{t}}(\mathbf{x}_j)\Delta t\right\} \sigma_{\mathrm{t}}(\mathbf{x}_i) L_s(\mathbf{x}_i) \Delta t$$

- Comparison against the general method --- DTRT [Zhang et al.]

DTRT, 1 spp, **7.5 ms**



DTRT, 256 spp, **1900 ms**

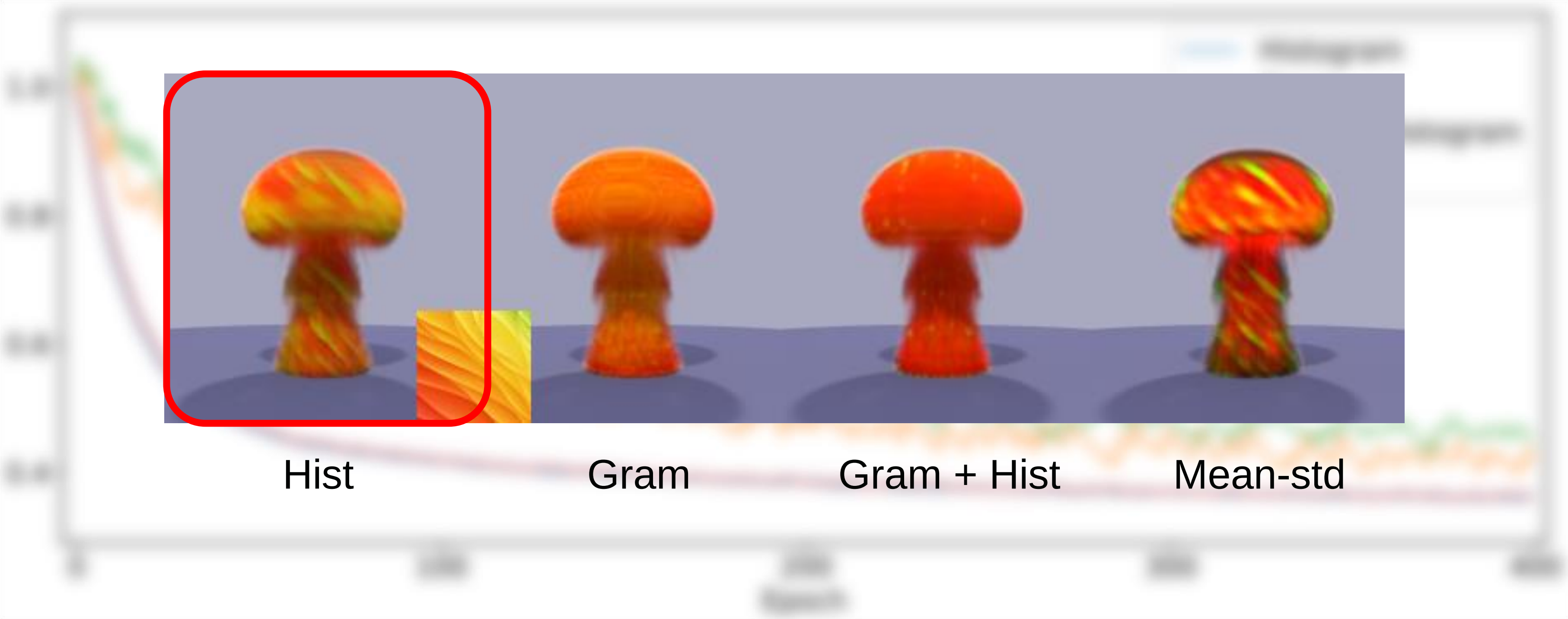


Ours, 1 spp, **0.3 ms**

LOSS FUNCTION

SIGGRAPH 2021

- Choice of style loss
  - Gram matrix, Histogram
  - Histogram + Gram matrix
  - Mean-std



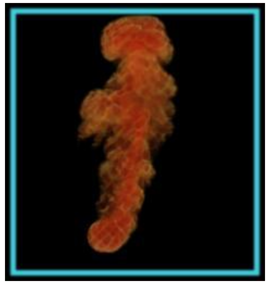
Hist   Gram   Gram + Hist   Mean-std

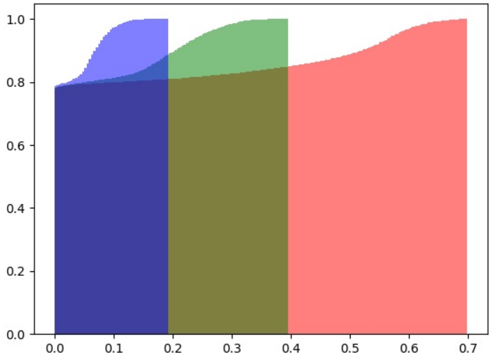
LOSS FUNCTION

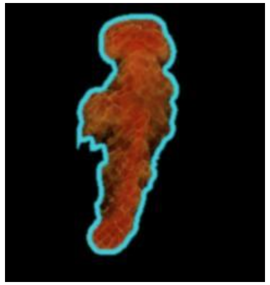
- Histogram loss
  - For a 2D image

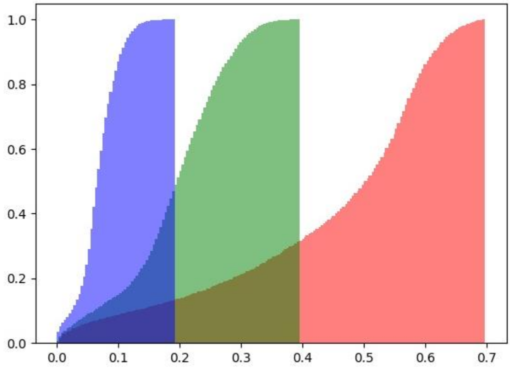
A mask for eliminating the influence of background

$$\mathcal{L}_{\text{hist-2D}}(\mathbf{I}, \mathbf{I}_s) = \sum_{l \in \{l_s\}} \sum_{c} \| \mathbf{M}^l \odot (\mathcal{F}_c^l(\mathbf{I}) - H(\mathbf{M}^l \odot \mathcal{F}_c^l(\mathbf{I}), \mathcal{F}_c^l(\mathbf{I}_s))) \|_2^2$$









  - For an albedo volume

$$\mathcal{L}_{\text{hist}}(\mathbf{V}_\alpha, \mathbf{I}_s) = \sum_{v \in \{v_p\}} \mathcal{L}_{\text{hist-2D}}(R(\mathbf{V}_\sigma, \mathbf{V}_\alpha, v), \mathbf{I}_s)$$

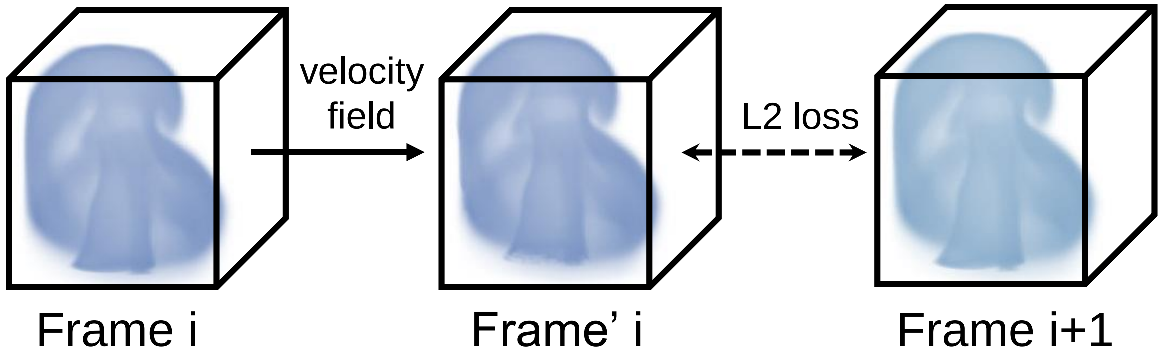
LOSS FUNCTION

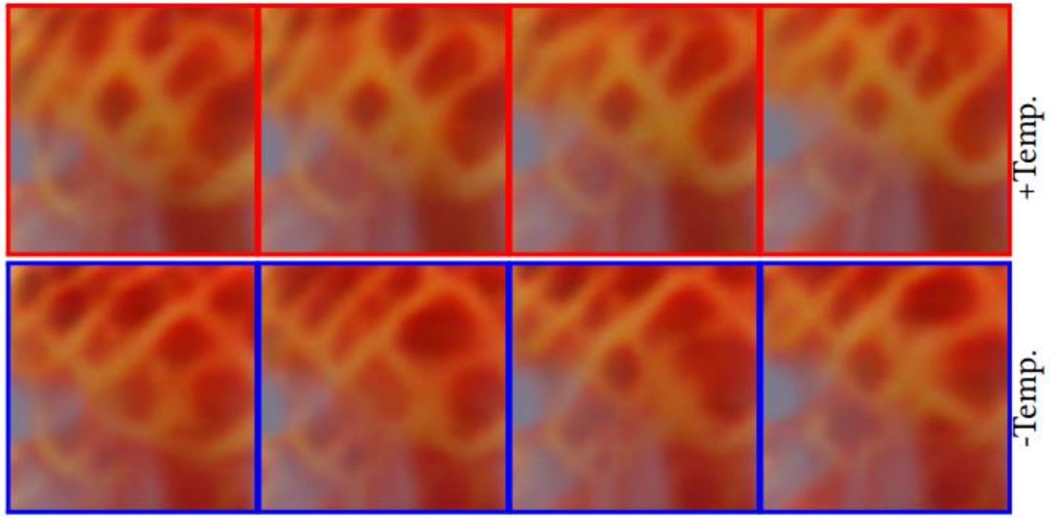
- **Total variation loss**
  - Encourage spatial smoothness in the generated albedo volume

$$\mathcal{L}_{\text{tv}}(\mathbf{V}_\alpha) = \|\nabla_x \mathbf{V}_\alpha\|_2^2 + \|\nabla_y \mathbf{V}_\alpha\|_2^2 + \|\nabla_z \mathbf{V}_\alpha\|_2^2$$

- **Temporal loss**
  - Penalize the inconsistencies between stylized volumes of two consecutive frames

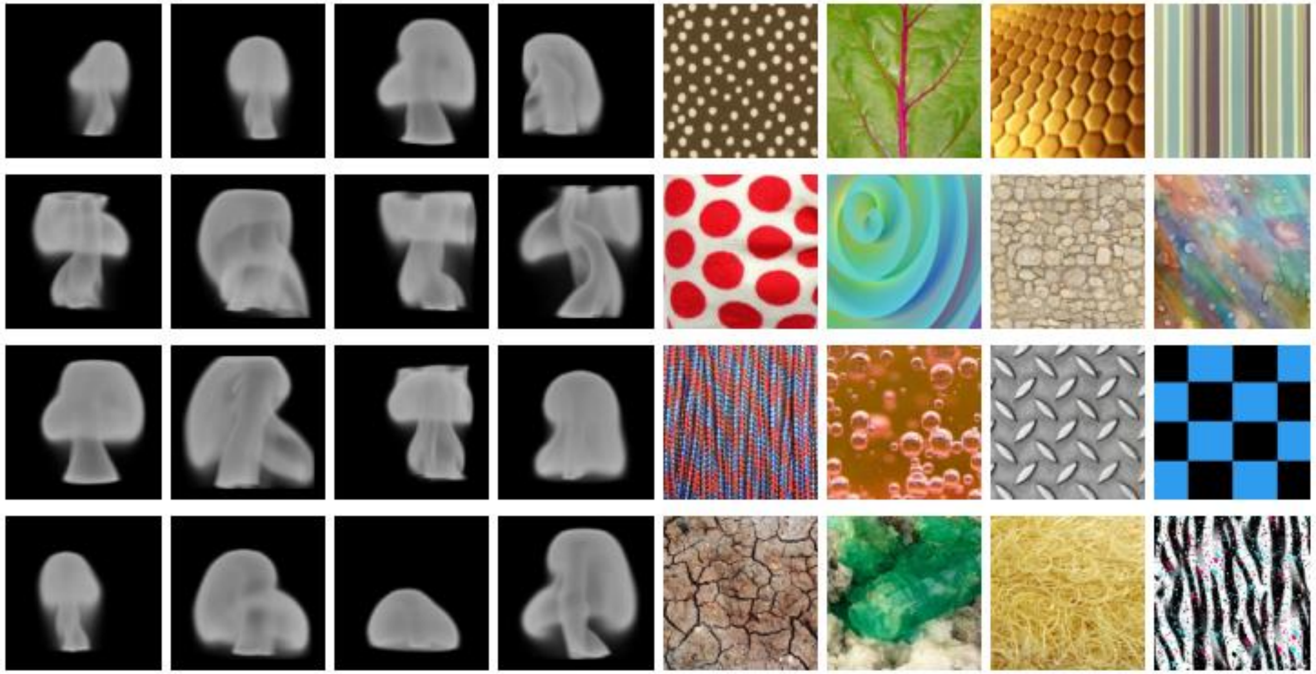
$$\mathcal{L}_{\text{temp}}(\mathbf{V}_\alpha, \mathbf{V}'_\alpha) = \|\mathbf{V}_\alpha - \mathcal{W}(\mathbf{V}'_\alpha, \mathbf{U})\|_2^2$$





DATASET

- 3k density volumes and velocity fields generated with *mantaflow*
- 3k style images from DTD dataset [Cimpoi et al.]



Density Volume

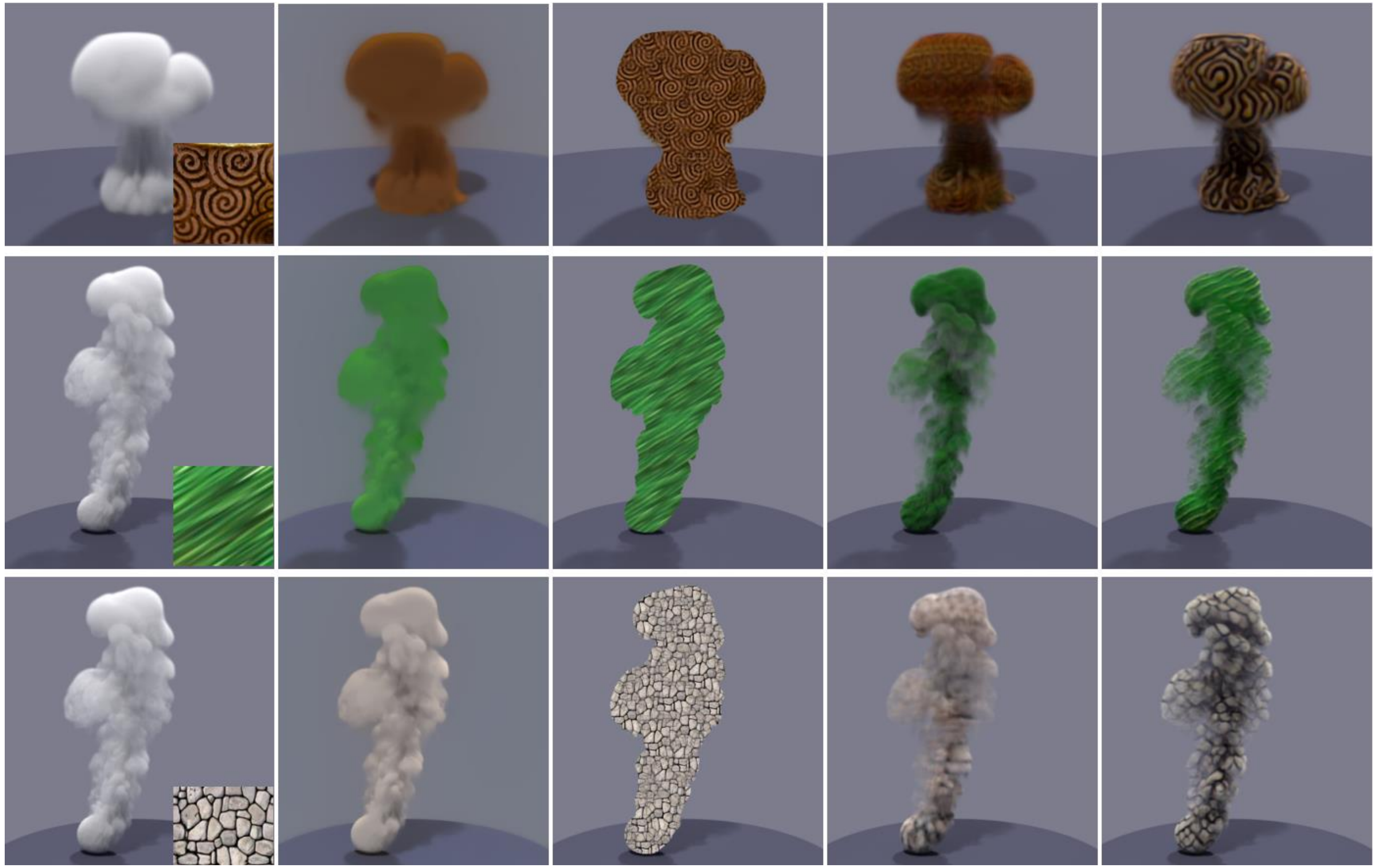
Style Image

COMPARISON WITH PREVIOUS METHODS

- We compare our method to
  - PhotoWCT [Li et al.] — a recent image style transfer method that is able to process arbitrary new style
  - LazyFluid [Jamriška et al.] — a 2D patch-based approach to appearance transfer for fluid animations
  - Tomography [Klehm et al.] — a tomographic reconstruction method for volumetric appearance stylization

COMPARISON WITH PREVIOUS METHODS

SIGGRAPH 2021



Input PhotoWCT LazyFluid Tomography SKPN (Ours)

Input PhotoWCT LazyFluid Tomography SKPN (Ours)

COMPARISON WITH PREVIOUS METHODS

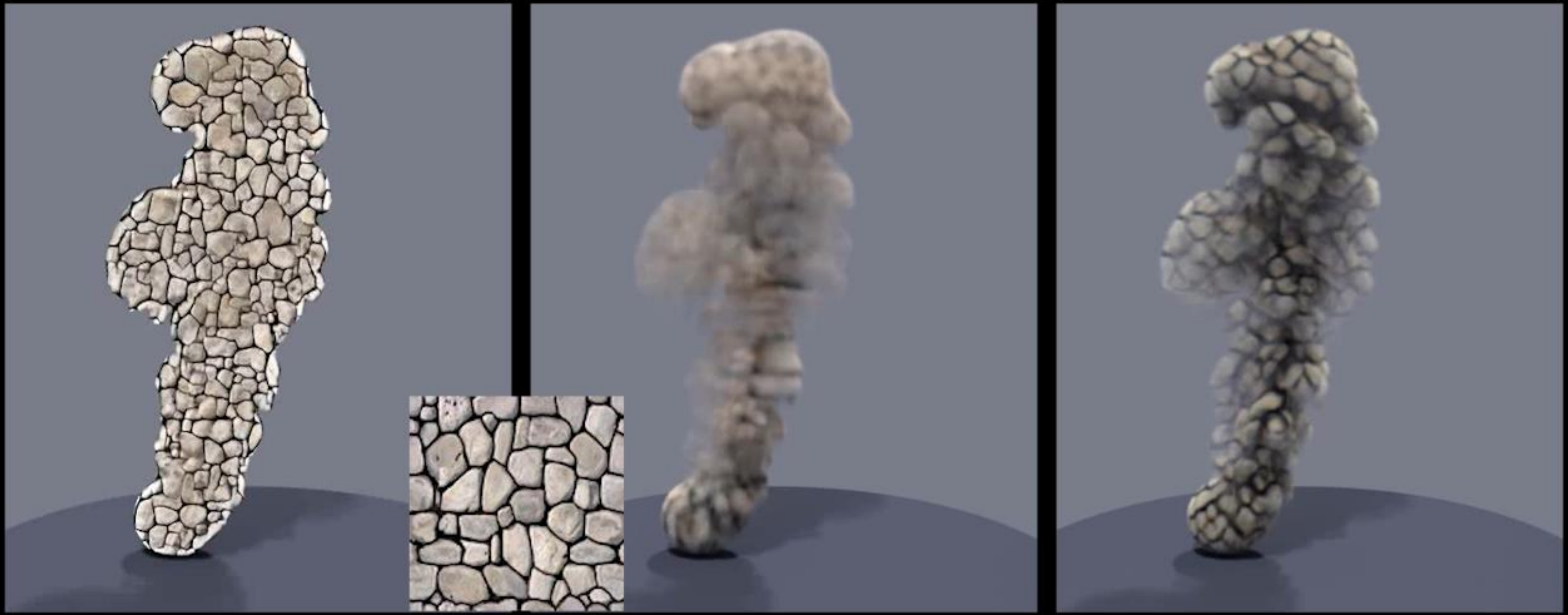
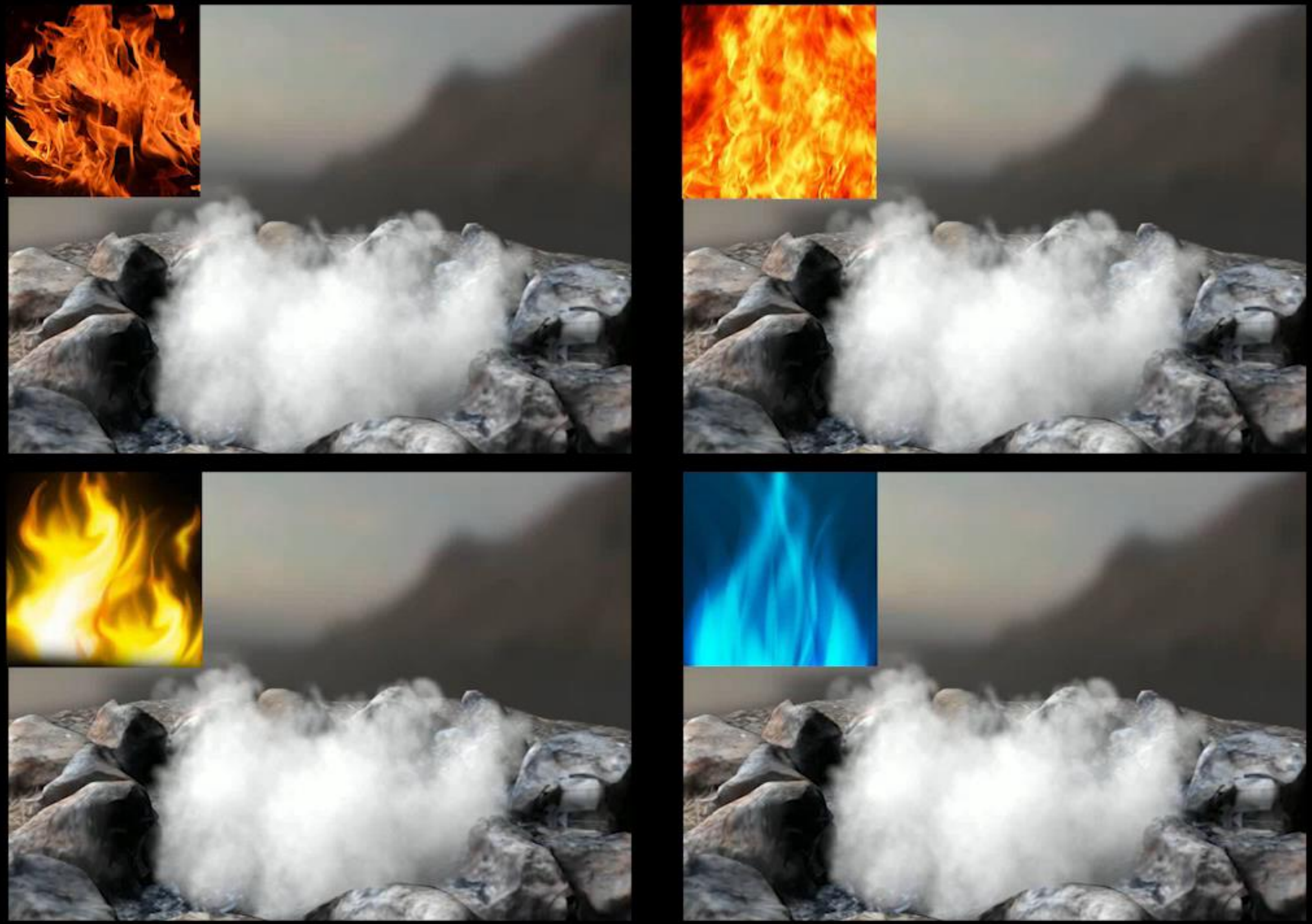


Image Style Transfer   Tomography   SKPN (Ours)

HALLUCINATING TRANSLUCENT MATERIALS

SIMULATING FIRE FLAMES



SIMULATING FIRE FLAMES





CONCLUSION AND FUTURE WORK

SIGGRAPH 2021

- Conclusion
  - We propose the first feed-forward neural network that enables efficient artistic volumetric appearance stylization
- Limitations
  - Tiny details can not be well captured
  - Can not guarantee consistency between color and structural features if adopt SKPN and TNST independently to stylize both density and albedo

Input TNST SKPN TNST+SKPN

SIGGRAPH 2021

THANK YOU !

Q&A