EUROGRAPHICS 2022 / R. Chaine and M. H. Kim
(Guest Editors)

Volume 41 (2022), Number 2

Deep Reconstruction of 3D Smoke Densities from Artist Sketches

Byungsoo Kim[1], Xingchang Huang[1,2], Laura Wuelfroth[1], Jingwei Tang[1], Guillaume Cordonnier[1,3], Markus Gross[1], Barbara Solenthaler[1]

[1]ETH Zurich [2]Max Planck Institute for Informatics [3]Inria, Université Côte d'Azur, France

1. Network Architecture

Our network architecture follows the U-Net architecture, similar to the updater CNN in [Delanoy et al. 2018]. The input consists of a reconstructed 3*D* density $\hat{d}$ and a 2*D* sketch s in a specific viewpoint, while the output is the residual of the refined 3D density field. Let conv2d, bn, lrelu, upsample denote 2*D* convolutions, 2*D* batch normalization, leaky ReLU activation (with slope 0.2) and nearest neighbor upsampling in each layer. Table 1 illustrates the details, such as the number of channels or kernel size. The skip connections shown as '+' represent the concatenation of two outputs with the same size. We use a linear activation in the last layer to predict the residual density $\hat{r}$, and the final output should be $\hat{d}+\hat{r}$ clamped between 0 and 1.

2. Training Data Generation

2.1. Data Set Generation in *Houdini*

We generate our data set with the *PyroFX* solver embedded in Houdini. In particular, we use the *Pyro Solver* node with parameters chosen randomly with a uniform distribution as depicted in Table 2 and the buoyancy direction (simulation/buoyancy_dir[x,y,z]) as a random unit vector.

2.2. Training Data Snapshots

Figure 1 shows additional selected training samples from our data set with different sourcing strategies. From left to right, the first four columns show *no source* examples, while the other columns depict *with source* scenes.

2.3. Sketch Variations

For the training, we augmented the sketches to increase the robustness of the model to (smaller) sketch variations. Specifically, we used the following sketch parameters, where $\mathcal{N}(x,\sigma)$ is the normal distribution centered at x and with variance σ:

$$\begin{aligned}
\theta_{bright} &= |\mathcal{N}(0,42.33)| \rightarrow [0,127] \\
\theta_{contrast} &= |\mathcal{N}(0,21.33)| \rightarrow [0,64] \\
\theta_{contour} &= 0.8 - |\mathcal{N}(0,0.1)|, \rightarrow [0.5,0.8] \\
\theta_{toon} &= 0.8 + \mathcal{N}(0,0.03), \rightarrow [0.7,0.9] \\
\theta_{lightdir} &= 1 + \mathcal{N}(0,0.66), \rightarrow [-1,3] \\
\theta_{blur} &= |\mathcal{N}(0,0.5)|, \rightarrow [0,1.5] \\
\theta_{slur} &= |\mathcal{N}(0,0.02)|, \rightarrow [0,0.06],
\end{aligned}$$

On the right side, the range of each parameter is given by 3σ. We augment different factors of sketch styles using Gaussian sampling. Corresponding figures for different styles are shown in Figure 2.

3. Initial Volume Modeling

Figure 13 illustrates the steps of the initial volume modeling for both front and side views of the smoke jet example. The image on the right corresponds to the input to our updater CNN.

4. Extended Results

4.1. Results on Artist Sketches

Figure 6, Figure 7 and Figure 8 show sketch-to-density reconstruction results from sketches drawn by 3 different artists. We observe that our approach is able to reconstruct density keyframes consistent to what artists have drawn. Reliable and robust reconstructions are a pre-requisite for interactive prototyping and authoring of smoke keyframes and animations.

4.2. Captured Smoke Data

We applied our method to the *Scalarflow* data set [Eckert et al. 2019] that contains reconstruction data of real-world captured smoke. We show the Scalarflow density and its sketch, and the reconstruction results (sketch and density) for scene ID 1 in Figure 9. The sketches have no notion of smoke thickness and therefore cannot distinguish between a wispy and a dense smoke volume with identical sketch shapes. The sketch loss therefore has a tendency to favor reconstructions that are less wispy than the ones of Scalarflow.

Table 1: *Details of the network architecture.*

Layer	Ops	K	S	P	Input	Output Size
input	-	-	-	-	density, sketches	$129 \times 129 \times 129$, 258×258
enc_1	conv2d+lrelu	4	2	1	sketch	$64 \times 129 \times 129$
enc_2	conv2d+bn+lrelu	4	2	1	enc_1+density	$256 \times 64 \times 64$
enc_3	conv2d+bn+lrelu	4	2	1	enc_2	$256 \times 32 \times 32$
enc_4	conv2d+bn+lrelu	4	2	1	enc_3	$512 \times 16 \times 16$
enc_5	conv2d+bn+lrelu	4	2	1	enc_4	$512 \times 8 \times 8$
enc_6	conv2d+bn+lrelu	4	2	1	enc_5	$512 \times 4 \times 4$
enc_7	conv2d+bn+lrelu	4	2	1	enc_6	$512 \times 2 \times 2$
enc_8	conv2d+lrelu	4	2	1	enc_7	$512 \times 1 \times 1$
dec_8	upsample+conv2d+bn+lrelu	3	1	1	enc_8	$512 \times 2 \times 2$
dec_7	upsample+conv2d+bn+lrelu	3	1	1	dec_8+enc_7	$512 \times 4 \times 4$
dec_6	upsample+conv2d+bn+lrelu	3	1	1	dec_7+enc_6	$512 \times 8 \times 8$
dec_5	upsample+conv2d+bn+lrelu	3	1	1	dec_6+enc_5	$512 \times 16 \times 16$
dec_4	upsample+conv2d+bn+lrelu	3	1	1	dec_5+enc_4	$256 \times 32 \times 32$
dec_3	upsample+conv2d+bn+lrelu	3	1	1	dec_4+enc_3	$256 \times 64 \times 64$
dec_2	upsample+conv2d+bn+lrelu	3	1	1	dec_3	$256 \times 128 \times 128$
dec_1	conv2d	2	1	1	dec_2	$129 \times 129 \times 129$

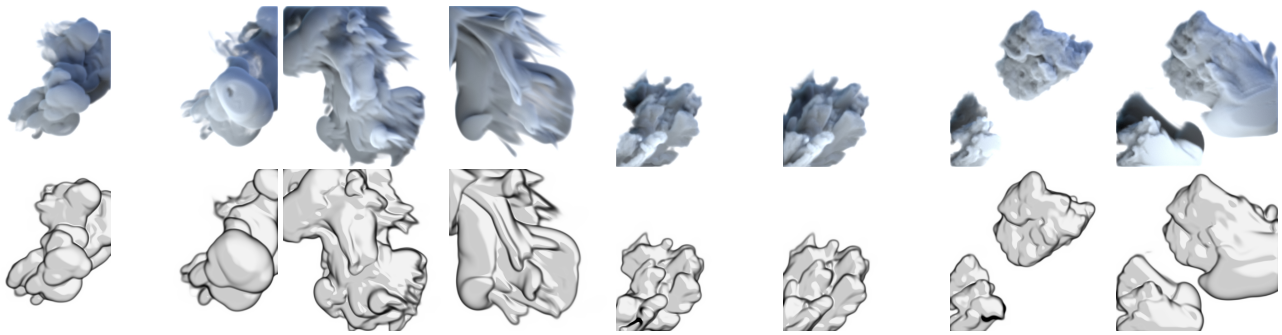

Figure 1: *Example snapshots from our training data set without (left) and with (right) source, showing density (top) and corresponding sketch computed with our sketcher (bottom).*

Table 2: *Simulation parameters used in Houdini.*

Tab	Parameter	Min value	Max value
Simulation	temp_diffusion (D)	0	.5
	cooling_rate	$.5 - 1.5\,D$	.8
	viscosity	0	.1
	lift	0	5
Shape	sharpenrate	0	1
	turbulence_scale	0	.1
Shape/Turb.	turb_swirl_size	.25	1
	turb_turb	2	5

4.3. Ablation Study: Loss Functions

We evaluated the impact of the different loss functions and illustrate the results in Figure 3. Using only the density loss (Equation 1) like in previous work [Delanoy et al. 2018] we observe large discrepancies between the input sketch and the sketch of the reconstructed density. When using only the sketch loss (Equation 2) results are very detailed but at the cost of depth ambiguity. If both density and sketch losses are used, the sketch correspondence as well as the depth reconstruction are improved, but noise is well.

4.4. Ablation Study: Multi-pass/view Refinement Results

We evaluate our network with various numbers of passes at training and test time. We show the results on front view only in Figure 10, where each line corresponds to 1, 2, 3 and 4 passes during training, respectively, and each column to successive refinements at test time that we denote by 'f' to 'ffffff'.

Our model is also able to refine density from multiple viewpoints. Figure 11 shows 6-view refinements ('fltdrb') on the smoke jet example of 3 models trained with $1, 2$, and 3 passes, respectively. Results are more robust for models trained with more passes during multi-view refinement. We observe that the difference between 3 and 4 passes is marginal, which is consistent to the 'ffffff' results shown in the paper and in Figure 10.

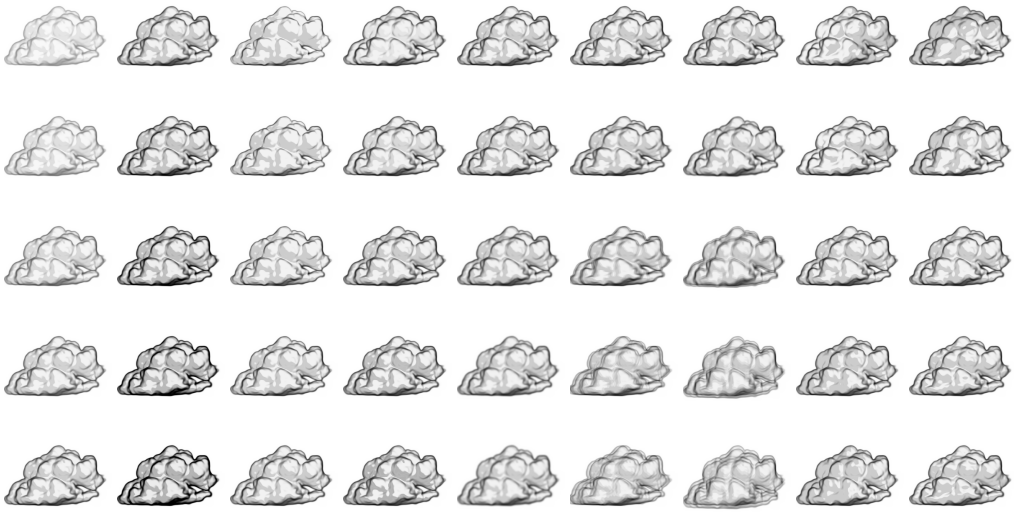

Figure 2: *Sketch augmentations are used for training to increase robustness. From left to right: brightness, contrast, contour width, toon shading color, blur in x, slur in x, slur in y, light direction in x, light direction in y.*

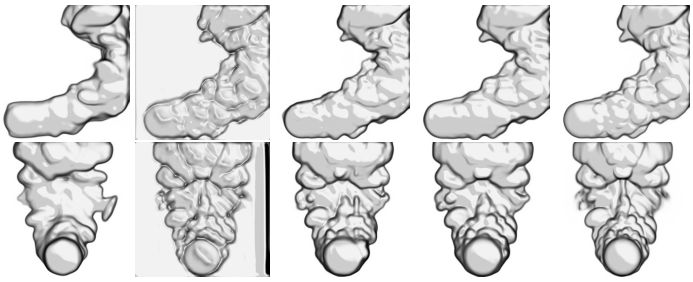

Figure 3: *Evaluation of loss functions. From left to right: density loss, sketch loss, density+sketch losses, density+sketch+depth variation losses (ours) and input sketch.*

4.5. Post-processing Network

We use a post-processing network to synthesize small-scale details from the sketches. In Figure 12 we show comparisons between results with and without post-processing steps.

4.6. Optimization-based Refinement

In the initial experiments we also experimented with optimizing a density field from input sketches using the method of [Okabe et al. 2015]. However, Figure 4 shows that in the generated results only sketches are matched, while the density converged to a non-optimal solution.

4.7. Convergence

Figure 5 shows the convergence of density and sketch losses during training.

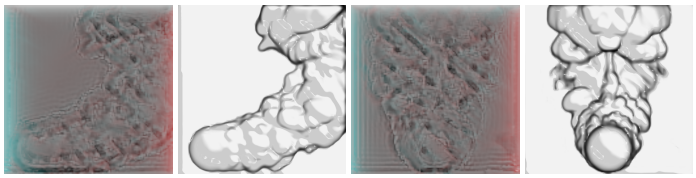

Figure 4: *Optimization-based sketch-to-density reconstruction results in non-optimal densities, where only the input sketches match. From left to right: front-view reconstructed density, sketch output and side-view reconstructed density and its sketch.*

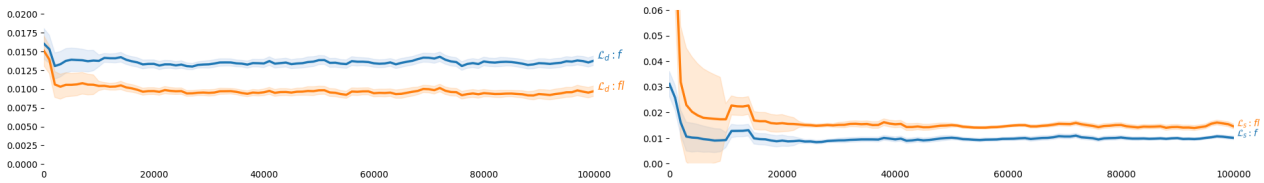

Figure 5: *Convergence plot of density (top) and sketch (bottom) losses during training for f (blue) and fl (orange). It shows that our model converges after* 100*k training iterations (i.e.,* 10 *epochs).*

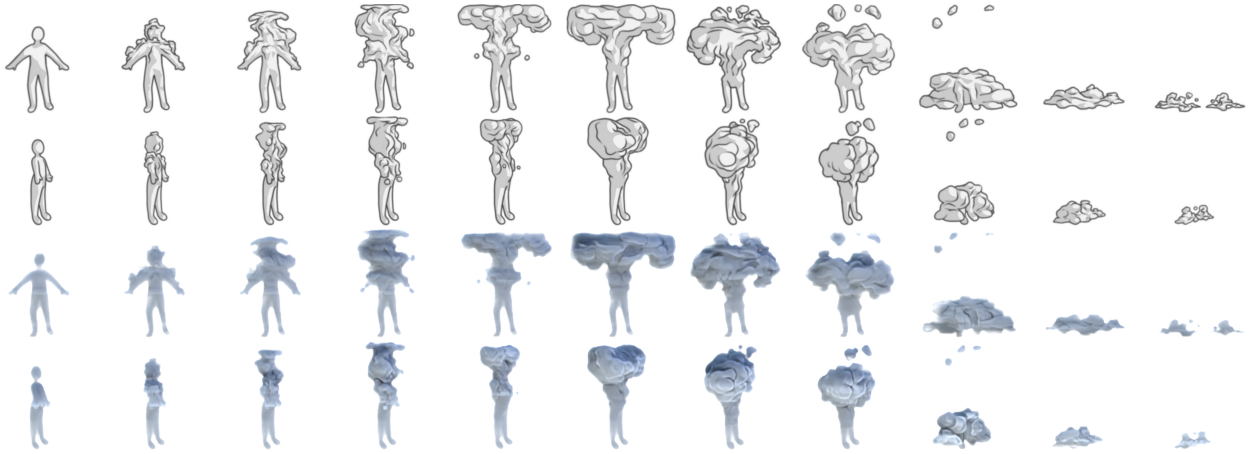

Figure 6: *Keyframe sequence (front and left views) of the* dissolve *scene drawn by an artist and corresponding 3D reconstructions.*

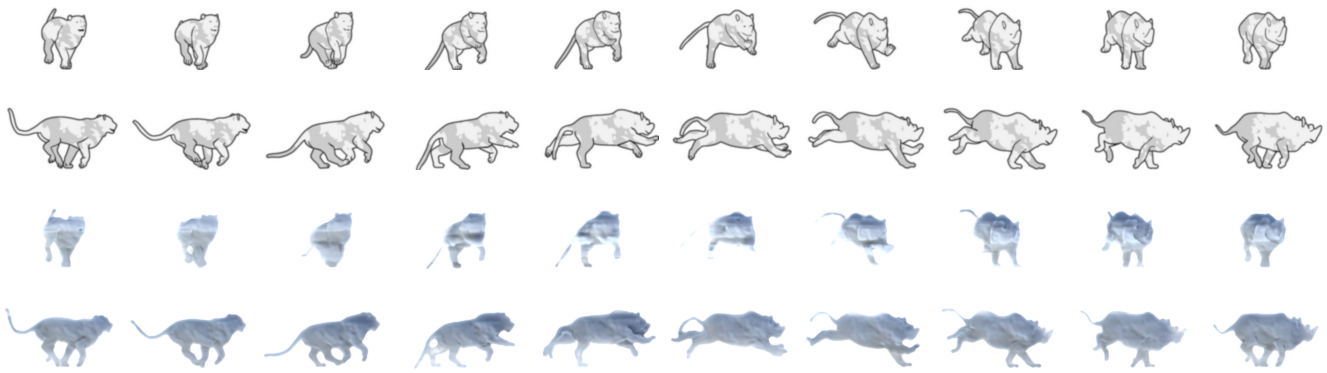

Figure 7: *Keyframe sequence (front and left views) of the* lion to rhino morphing *scene drawn by an artist and corresponding 3D reconstructions.*

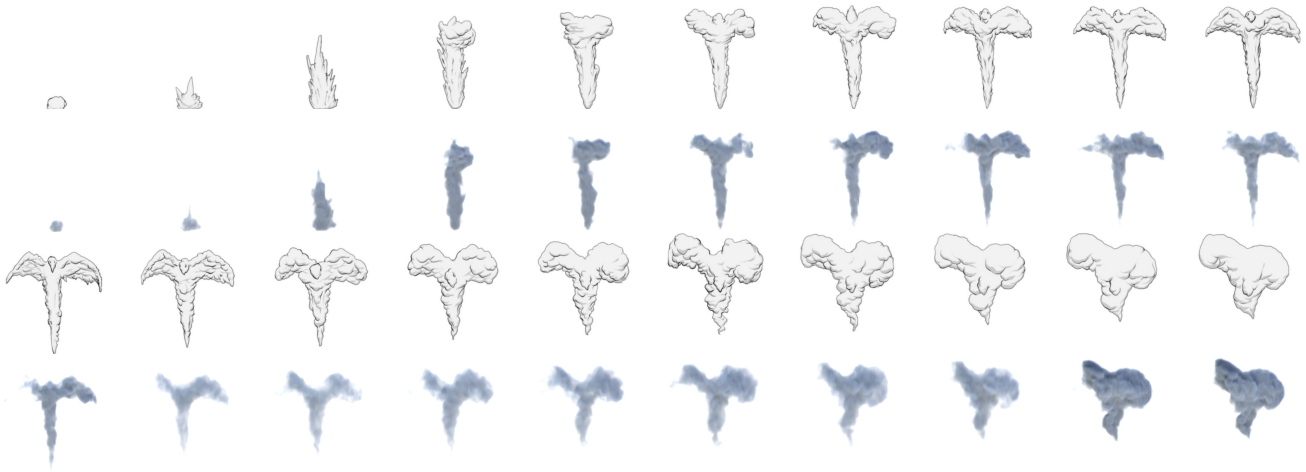

Figure 8: *Keyframe sequence (front view) of the* bird *example sketched by an artist and corresponding 3D density reconstructions.*

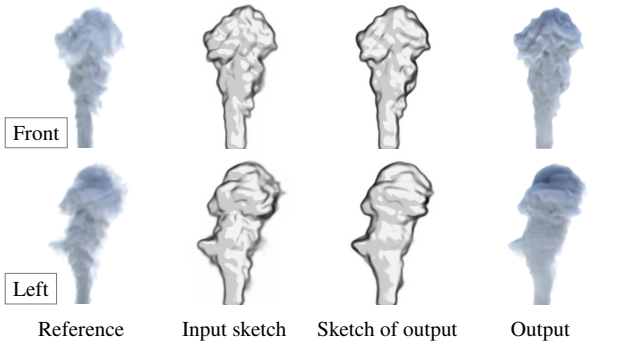

Figure 9: *Front (top) and left (bottom) views of a selected example from the* Scalarflow *data set. From left to right: Scalarflow density, input sketch, reconstructed sketch and output density.*

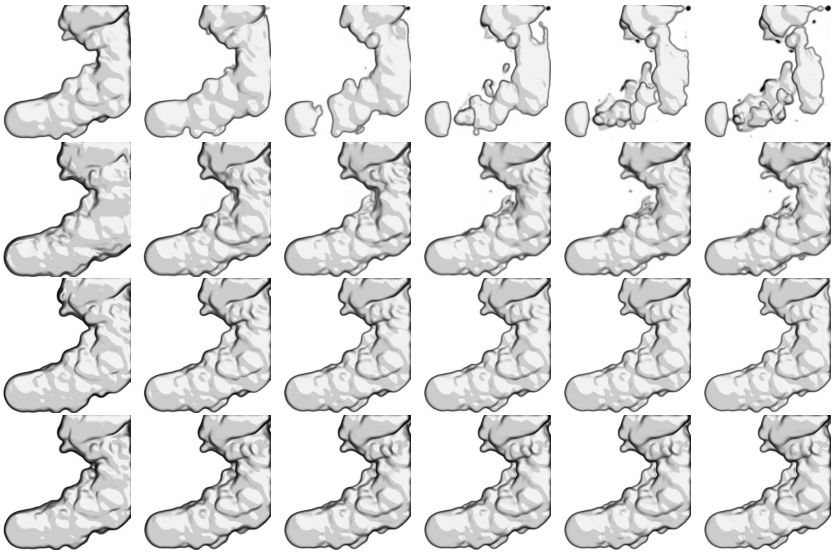

Figure 10: *Evaluation of recursive passes and inference sequence. Top and bottom rows show 1 and 4 passes, respectively. From left to right: f, ff, fff, ffff, fffff, ffffff*

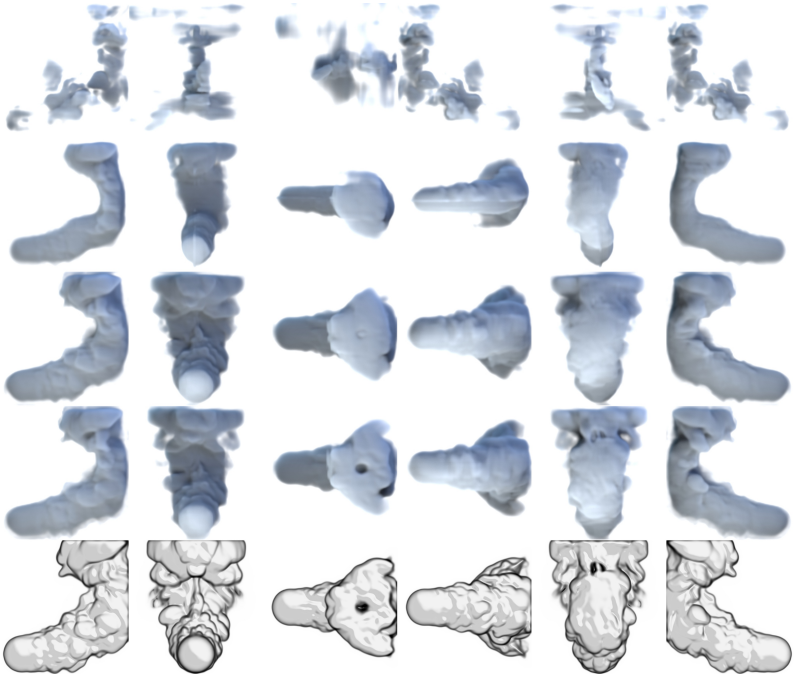

Figure 11: *Evaluation of multi-view refinements. Top to bottom rows show 6-view refinements on models trained with* $1, 2, 3, 4$ *passes and input sketches. From left to right:* 6 *views (front, left, top, bottom, right, back).*

Figure 12: *Comparisons between results without and with post-processing. The top and bottom rows of each example show the front and side view, respectively.*

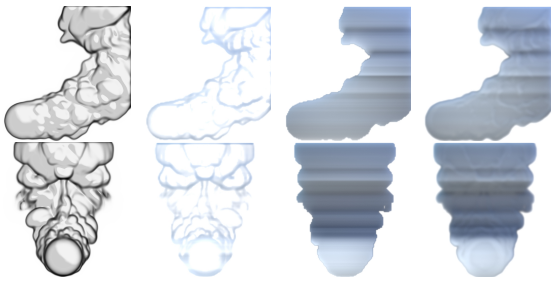

Figure 13: *Steps of the initial volume modeling shown for the front (top) and left (bottom) views. From left to right: input sketch, contour extraction, volume estimate, blended and smoothed volume (input to the updater network).*