Supplementary Material

7.1. Implementation Details for Cubic Bézier Curves

Our sampler is implemented for piece-wise cubic Bézier curves as it provides a flexible and computationally efficient representation. In our experiments, along with manually created curves using Inkscape and parsed using open source nanosvg [nan], we also used scanned hand-drawn sketches that are vectorized using [FLB16] since curve identification is not part of our contribution. Our sampling does not require any tangent continuity between the piece-wise cubic Bézier curve segments. However, using C^1 continuity will avoid the vanishing of the local feature size at joints, leading to better results, though, for our applications, we show that we can handle these artifacts as well.

The sampling process on Bézier curve segments is performed by inserting foot points as follows. Let $p \in C$ correspond to a point evaluated as $B_k^3(t)$ on curve segment k at parameter value t; for simplicity, we will denote this using the notation $B(t)$. Each foot point $f_i \in C$ can be evaluated as $f_i = B(t)$. The normal $n(t)$ to C at a point $p = B(t)$ is orthogonal to the tangent $B'(t)$ at p, thus $n(t) = [B'_y(t), -B'_x(t)]^T$. To compute the medial point m_i for a foot point $f_i \in C$, we determine m_i as the center of a disk D_i with radius r_i. This disk is tangent to C at f_i and constrained by passing through the subsequent foot point f_{i+1}, thus fixing r_i and, later, m_i. As m_i may be located on either side of the curve, we first initialize our radius estimate r_i with a circle of radius equal to the bounding box diagonal of C. For both sides, we then compute the closest point q of the curve (for all segments) to the current medial point estimate, and if it is closer, replace it with the center of the circle tangent to C at f_i and passing through q. We iterate until a desired precision threshold (10^{-9}) is reached and select the point m_i from the side that is closer to f_i. Note that this threshold always converged in our experiments but could be further adjusted proportionally to curve sampling distances.

The distance of a point m to a cubic Bézier curve B can be expressed as $|B(t) - m|$. We translate the coordinates of B such that m is at the origin. Then, in order to find the point on B closest to m, $B(t)$, we minimize the squared term $|B(t)|^2$, thus eliminating the square root, by setting its first derivative $(|B(t)|^2)' = 0$. This results in a quintic polynomial that we solve using a specialized quintic root finder [qui] that employs the real roots isolation method using both Cauchy's bound as well as Kojima's bound, as it is several times faster than the general Eigen library method (see Sec. 3.4 for the analysis). Since computing the closest point to a cubic Bézier curve is still expensive, we first test its bounds using its convex hull property. Testing whether any point of a convex hull edge is closer than the current minimum distance, using the same edge projection procedure as in Sec. 3.2, accelerates it further, resulting in a global runtime reduced by an order of magnitude, as shown in Table 2.

Uniform meshing						
Density	0.1	0.05	0.03	0.02	0.015	0.01
Triangles	1087	1654	3373	6702	11333	24536
Runtime	0.011	0.021	0.069	0.273	0.835	6.784
RMSE	3.526	3.106	2.700	1.933	1.703	-
Uniform meshing						
Triangles	249	252	267	301	350	477
Runtime	0.013	0.012	0.014	0.013	0.016	0.074
RMSE	7.631	3.481	2.039	2.354	1.268	-

Table 4: *Sampling densities evaluated for the simulation w.r.t. 0.01: Density is maximum sampling distance/edge length in bounding box diagonal, runtime in seconds, and RMSE in terms of the uniform Dirichlet boundary conditions, for the five sketches' average.*

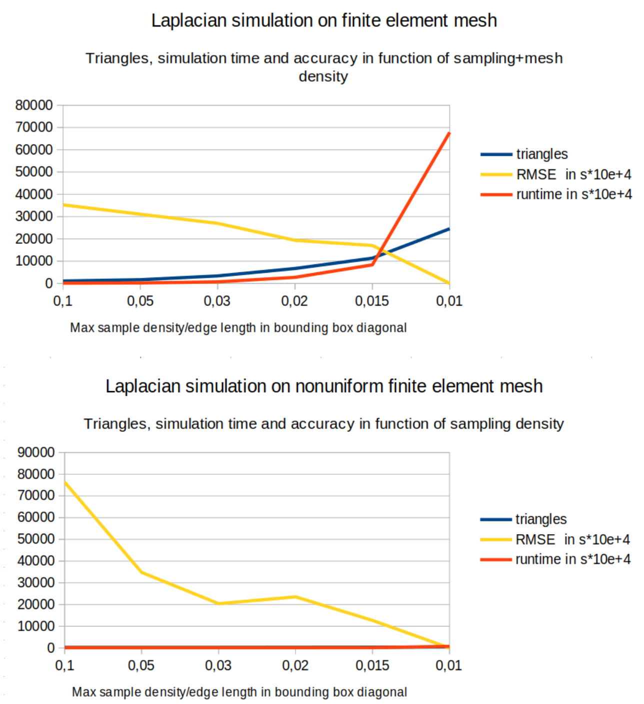

Figure 10: *For varying sampling densities (=maximum edge length of the meshed triangulation), triangle count, runtime and RMSE (compared to 0.01 density) of the Laplacian simulation are shown.*

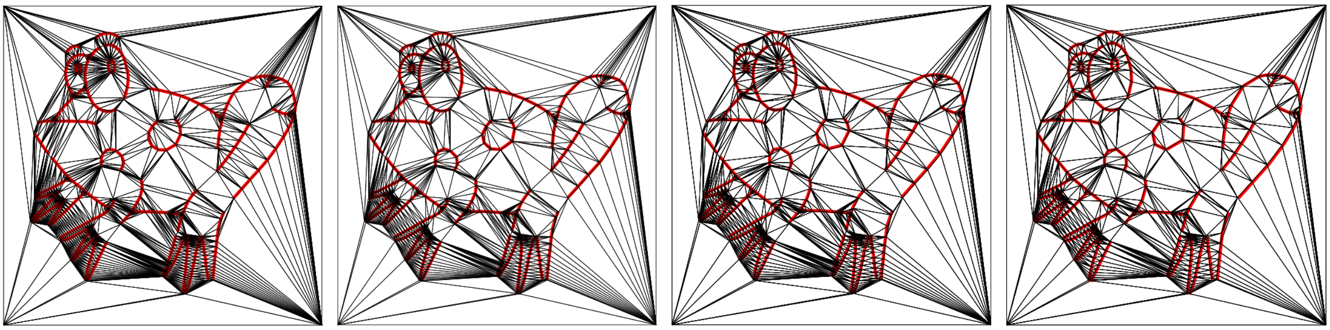

Figure 11: *Delaunay conforming property with different* ε *values. Left to right:* ε *= 0.4, 0.5, 0.6, 0.7*

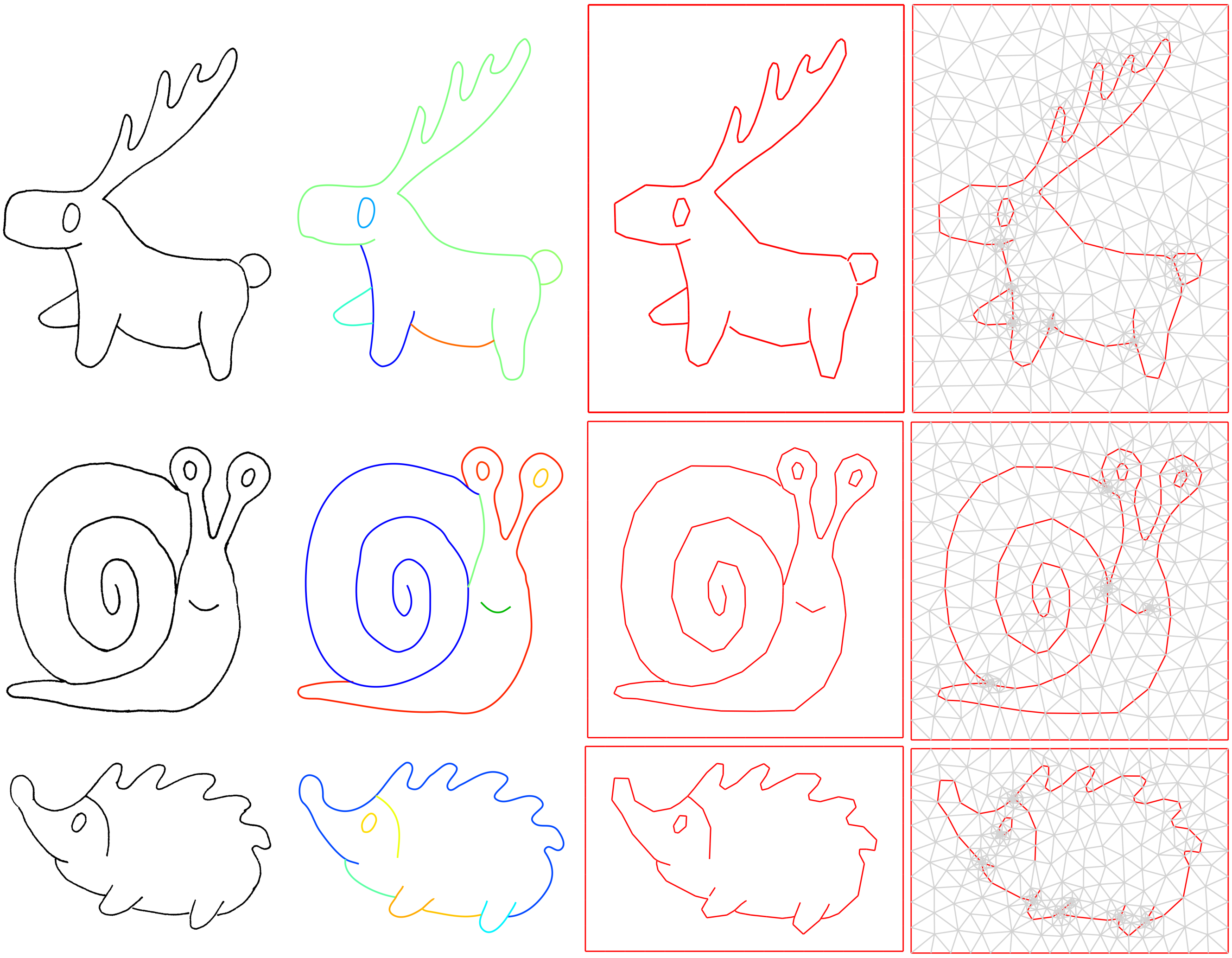

Figure 12: *From left to right: Scanned user sketch, after fitting Bézier curves, sampled with* $\varepsilon = 1$ *constrained to [0.01, 0.1] distance in terms of the bounding box diagonal, and the result of feature-aware meshing with max edge length = 0.1.*

Proceedings published by Eurographics - The European Association for Computer Graphics.

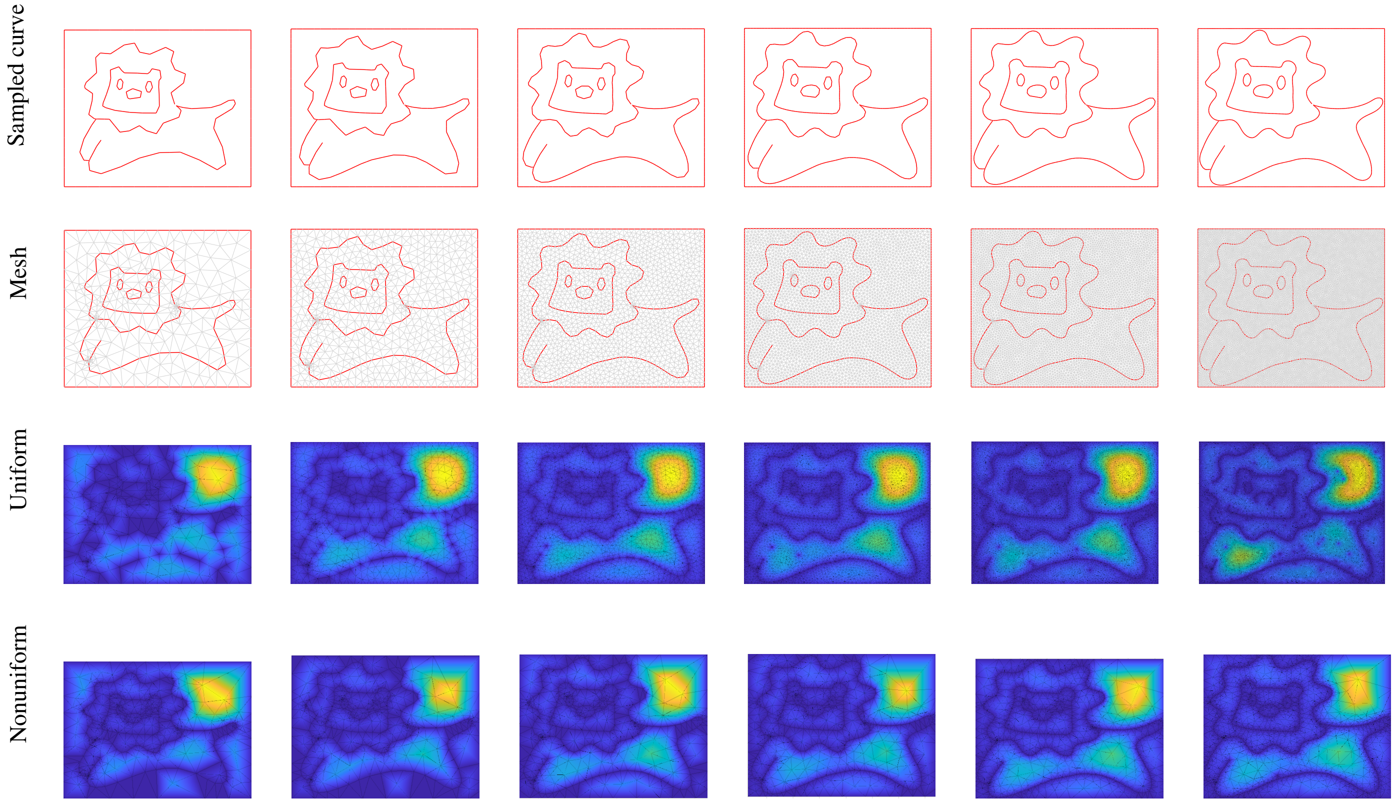

Figure 13: *Left to right: Varying sampling densities of 0.1, 0.05, 0.03, 0.02, 0.015, and 0.01 of bounding box diagonal for the* LION. *Top to bottom: Sampled curve connected by polylines, Meshed triangulation, and Visualization of the Laplacian simulation results with the Dirichlet conditions as dark blue dots from the vertices of the above polylines as well as the rectangular boundary, above with uniform and then below with non-uniform meshing.*

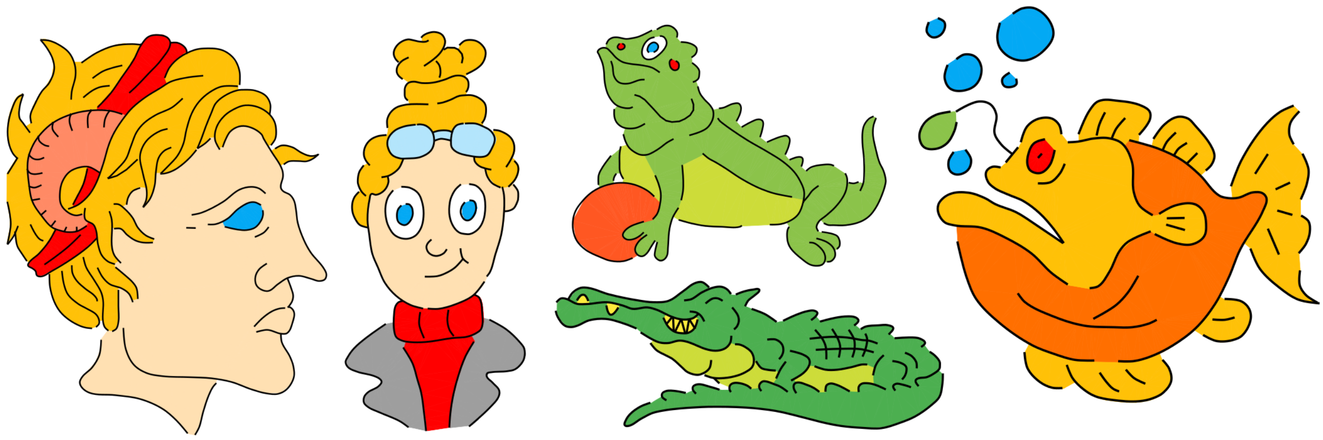

Figure 14: *Vector sketches colored using our improved Delaunay Painting [PMC22] - Images taken and vectorized from [PMC22]*