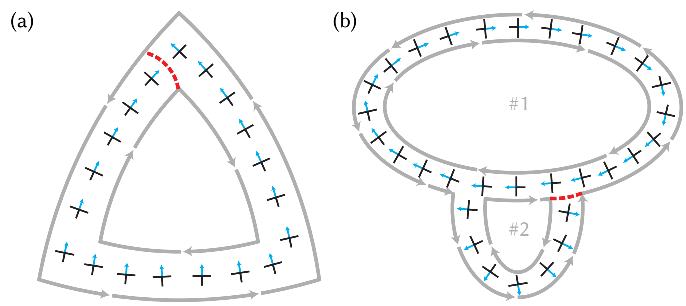
Supplementary Material for "Singularity-Free Frame Fields for Line Drawing Vectorization"

Olga Guțan, Shreya Hegde, Erick Jimenez Berumen, Mikhail Bessmeltsev, Edward Chien

1 Boundary Index Examples

Here we provide two examples of boundary indices for simple domains and cross fields. Note that the boundary indices constrain the number of "corners" implied by the cross fields.



$\chi = 0$	Ext	Int	$\chi = 1$	Ext	Int_1	Int_2
Cpt rotation	$\frac{\pi}{2}$	$-\frac{\pi}{2}$	Cpt rotation	3π	-2π	$-\pi$
Rep rotation	2π	-2π	Rep rotation	12π	-8π	-4π
Index	0	0	Index	-5	3	1

Figure 1: Some examples of simple narrowband domains with cross fields and boundary indices. It is often easier to figure out the component vector rotation and to multiply it by 4 to obtain the representative vector rotation. A particular component vector has been highlighted to assist in this, and red dashed lines denote places where there is a discontinuity in the matching. Note that the sum of indices in both cases is equal to χ.

2 Field Combing Examples

Recall that the boundary indices roughly inform the number and type of sharp turns in the vectorized curves to-be-traced from our frame fields. In the examples below, we show local pictures of initial convex cross fields, their singularities, and their matchings to each other or boundaries. We also comment on the effect of this combing.

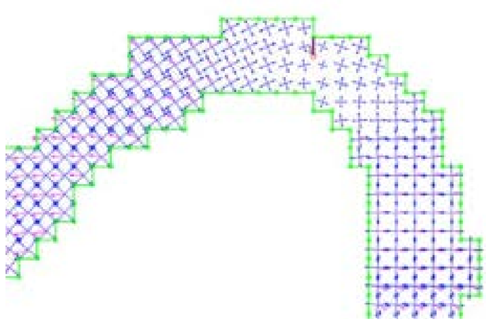


Figure 2: crop from "donkey", top of eye

In the example above, our initial crossfield has a -1 singularity in the interior. This is resolved by pushing it to the upper boundary, resulting in the desired smooth curve at that point, instead of a 90 degree turn.

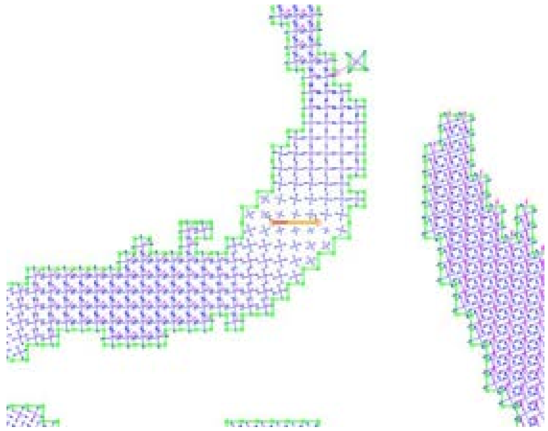


Figure 3: crop from "donkey", bridge of nose

In this example, our initial crossfield has 2 singularities of opposite signs in the curve interior. These are matched to each other, ensuring that the curve is traced smoothly. Note that simply pushing these singularities to the boundaries would result in a 90 degree turn and a "corner" at this point.

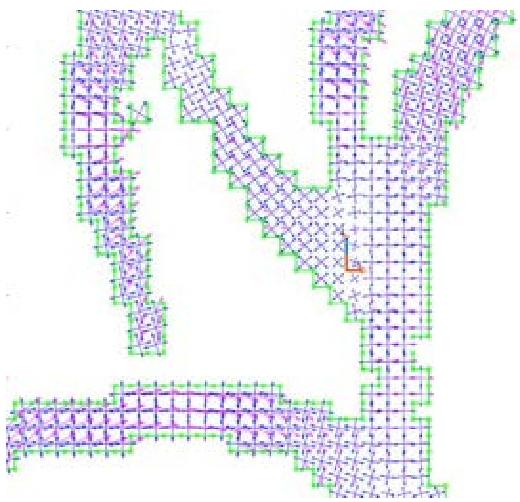


Figure 4: crop from "rose"

In this last example, we again see two internal singularities matched with each other in a junction. This merging ensures that this junction remains a Y-junction, and simply pushing these singularities to their closest boundaries would turn this into an undesirable T-junction.

3 Additional Vectorization Comparisons

We include here a gallery of additional comparisons using the tracing method of [BS19]: when using the original PolyVector fields in [BS19] (left, red) and when using our improved frame fields (right, green).

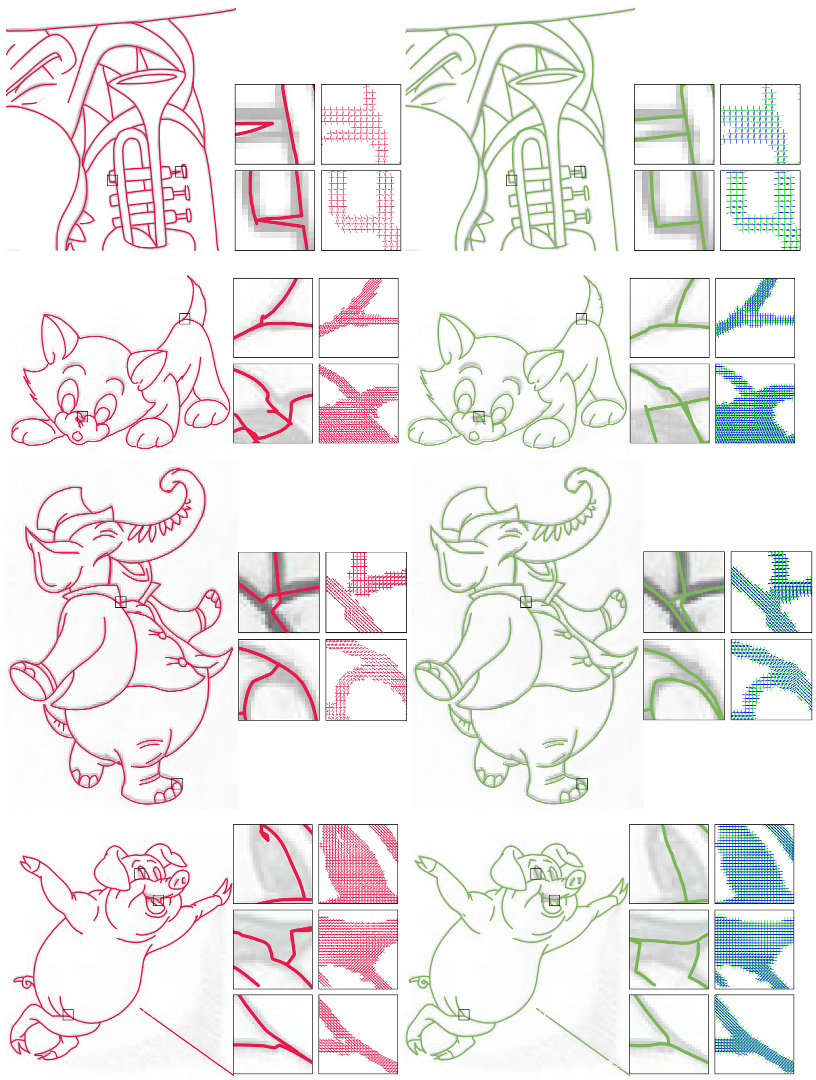


Figure 5: Our method disambiguates junctions more robustly (kitten), and infers the correct topology (trumpet). This yields higher-quality vectorizations that are closer to the intended drawn image (pig). Inputs from [Noris et al. 2013] (trumpet) and https://www.easy-drawings-and-sketches.com/ ©Ivan Huska (others).

4 Comparison on Cross-Hatching

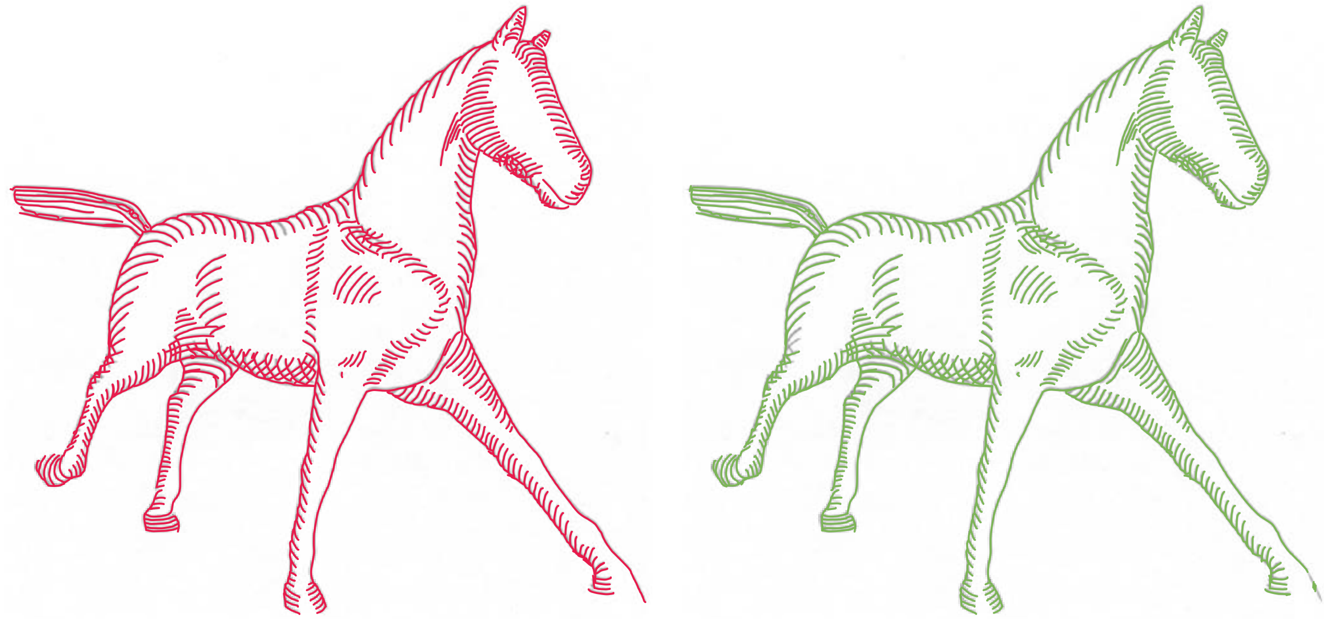


Figure 6: In addition to line drawings, our method (green curves) can be used to vectorize drawings in the cross-hatching technique, where at most points there are only two curves crossing. We provide the vectorization generated using polyvector fields on the left (red curves). Input drawing by Olga Vesselova [KNBH12]

5 Shared Contrast Degradation

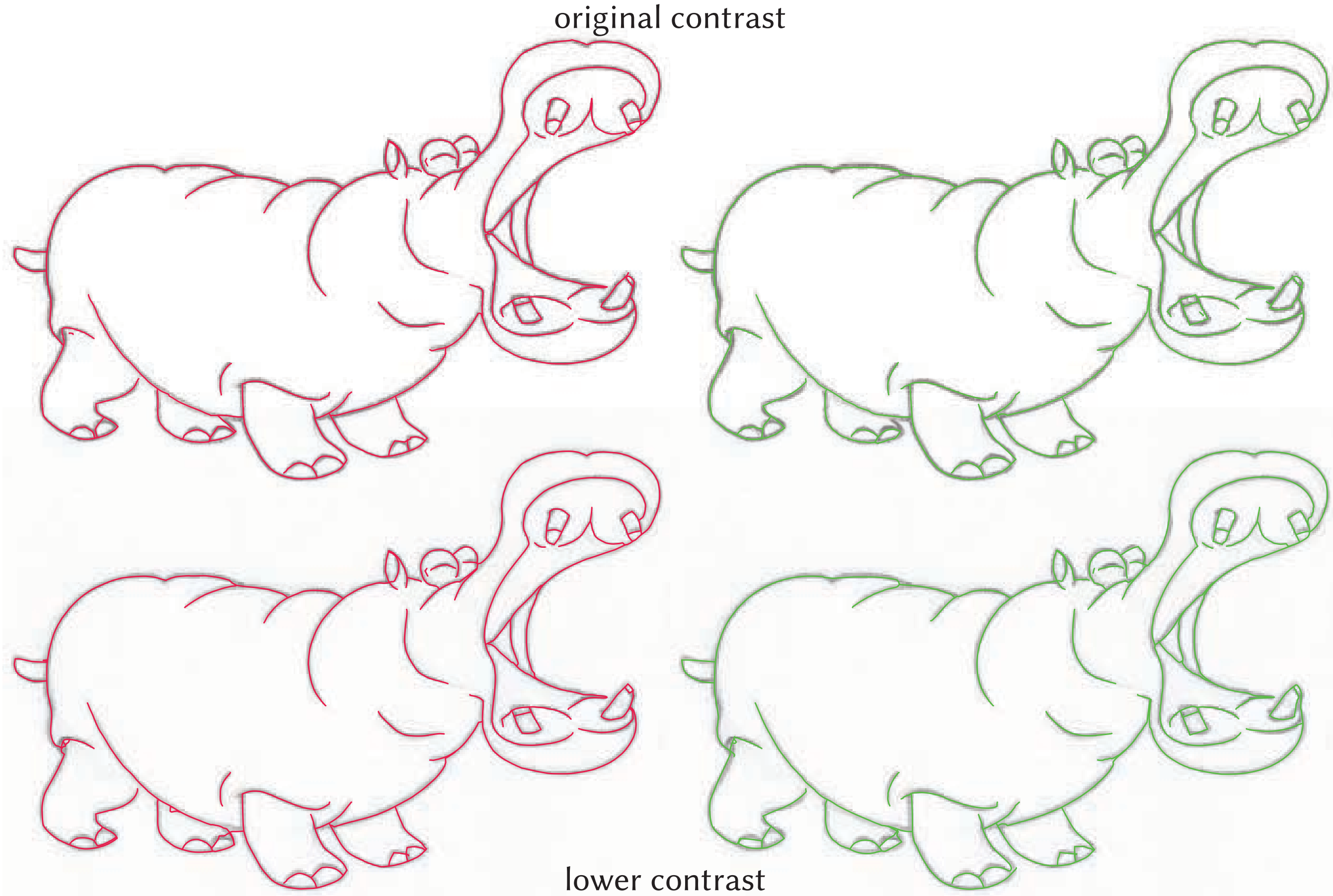


Figure 7: Both the [BS19]'s and our method's results degrade when the input image contrast is lowered.

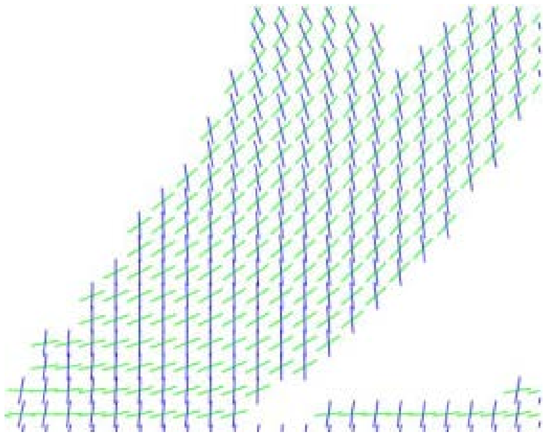
6 Quantitative Comparison to Prior Methods

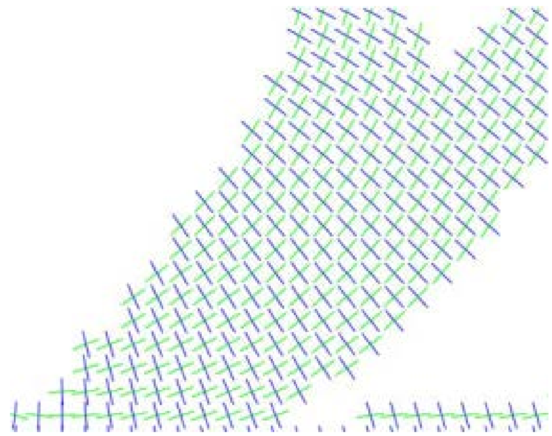
We compute the Chamfer distance, using the Euclidean distance as our metric. Our results are quantitatively extremely similar to those of [BS19] or [PNCB21], depending on the method used, thus we do not run our algorithm on the benchmark of [YVG20].

	Chamfer	Distance
	[BS19]	[PNCB21]
Cup	0.001	0.0009
Sketch1	0.24	
Sketch2	0.2983	
Donkey	0.1733	
Elephant	0.2107	
Goldfish	0.0446	0.0543
Hippo	0.1053	0.1877
Horse	0.1354	
Kitten	0.1481	
Pig	0.1897	
Rose	0.0186	
Trumpet	0.0527	

Table 1: Chamfer Distance

7 Anti-Align Effect

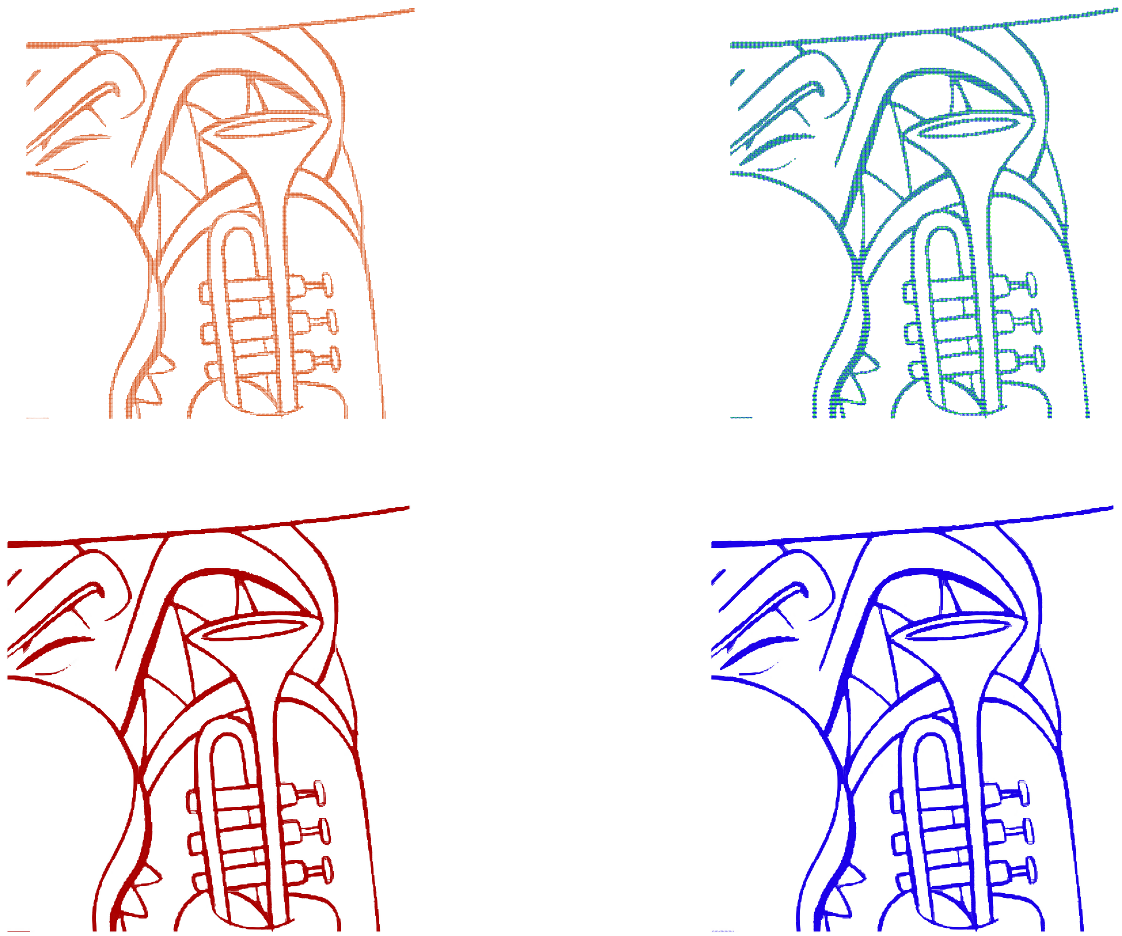


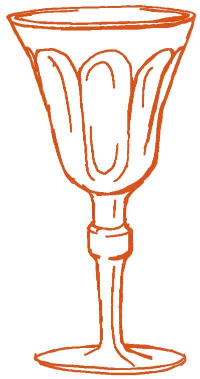


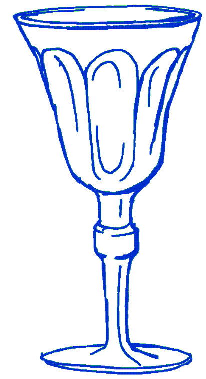
In the above images, we show the effect of the anti-align energy term, E_ν of Equation (13). On the left, when there is no anti-align term, the field settles into a local minimum where both axes of the field are partially aligned to the true tangent. This will throw off the tracing procedure and produce incorrect tangent behavior and topology for the resulting vectorization curves. On the right, we can see that this field behavior is corrected with inclusion of the anti-align term.

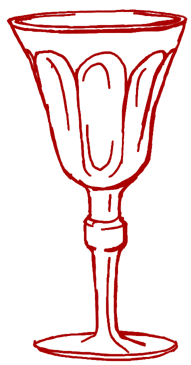
8 Frame Fields and Streamlines Comparisons

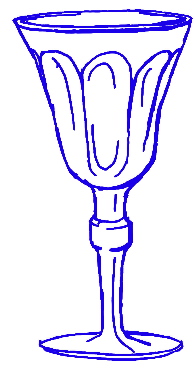
Below on the next few pages, we present high-quality comparisons of the frame fields and streamlines for results shown in the paper. The PolyVector fields, designed in [BS19] and used both in [PNCB21] and [SBBB20] and the corresponding streamlines are on the left in red; our fields and streamlines are on the right in green and blue.

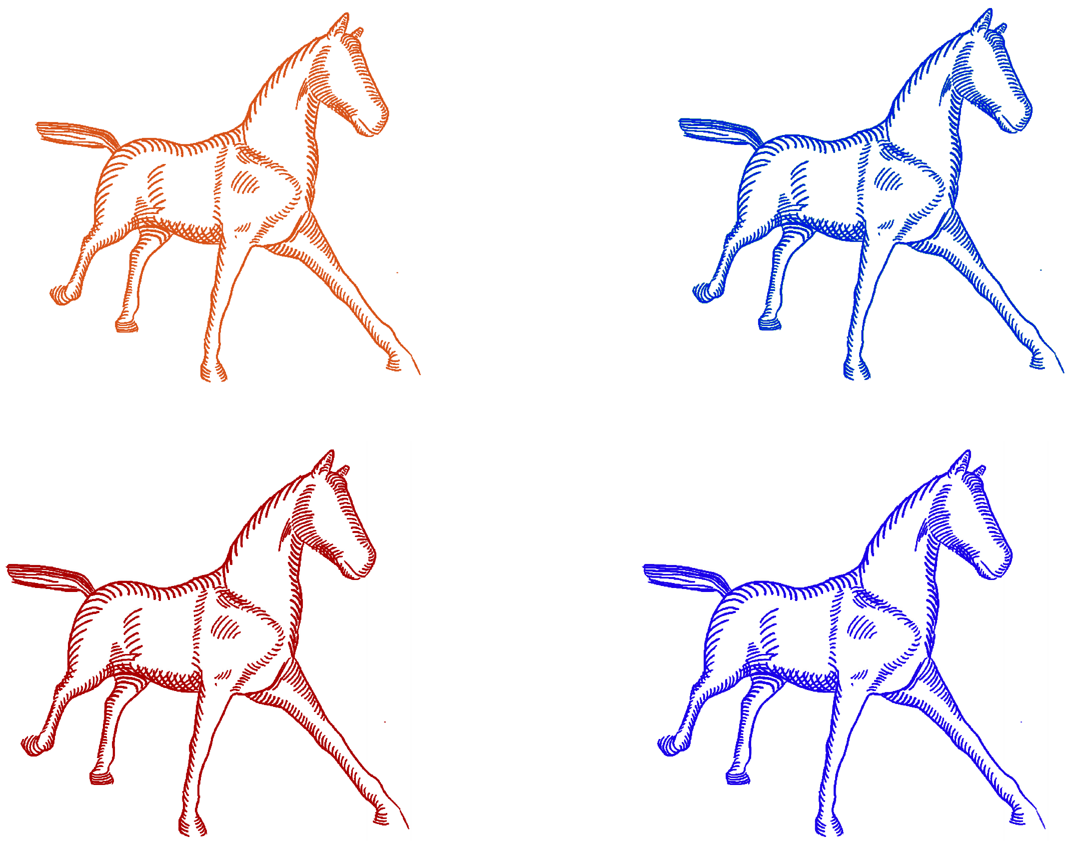


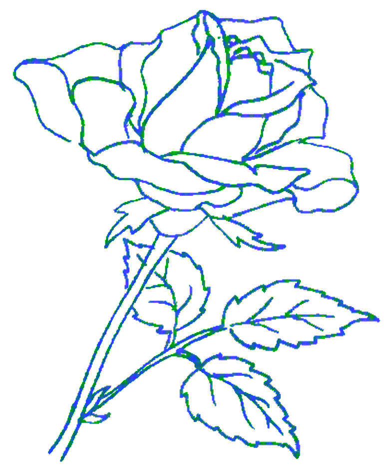


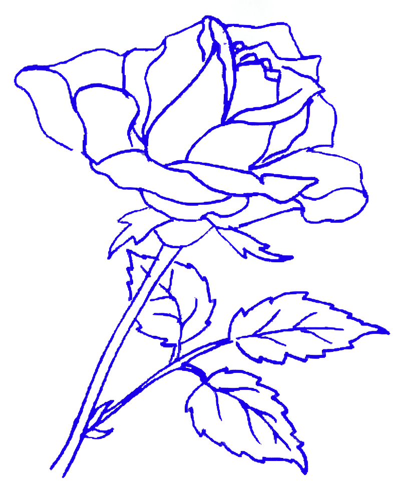


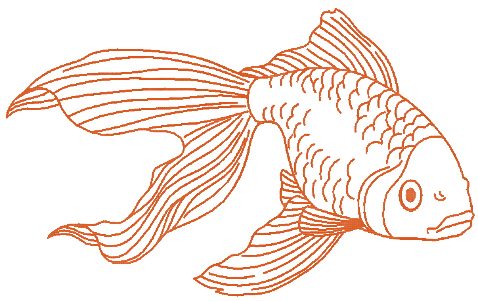
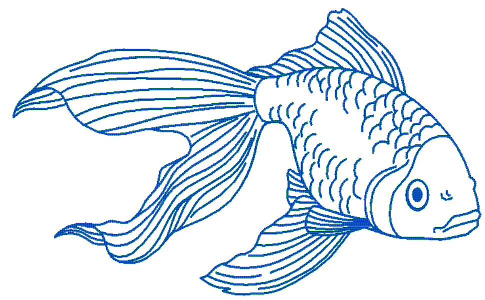
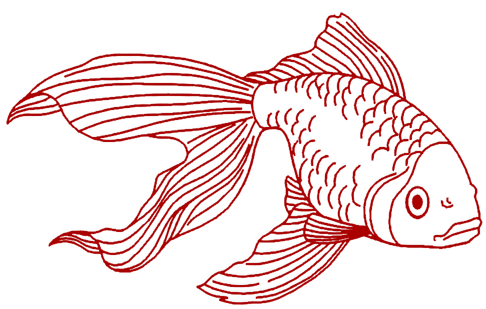
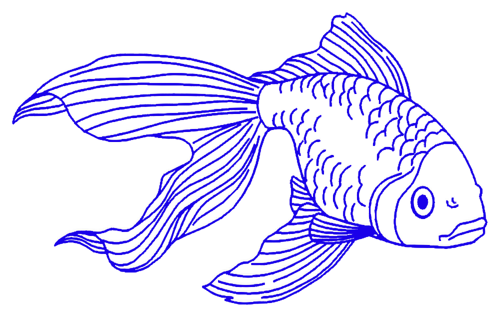











References

[BS19] Bessmeltsev M., Solomon J.: Vectorization of line drawings via polyvector fields. *ACM Trans. Graph. 38*, 1 (2019).

[KNBH12] Kalogerakis E., Nowrouzezahrai D., Breslav S., Hertzmann A.: Learning hatching for pen-and-ink illustration of surfaces. *ACM Trans. Graph. 31*, 1 (2012).

[PNCB21] Puhachov I., Neveu W., Chien E., Bessmeltsev M.: Keypoint-driven line drawing vectorization via polyvector flow. *ACM Trans. Graph. 40*, 6 (2021).

[SBBB20] Stanko T., Bessmeltsev M., Bommes D., Bousseau A.: Integer-grid sketch simplification and vectorization. *Computer Graphics Forum 39* (2020).

[YVG20] Yan C., Vanderhaeghe D., Gingold Y.: A benchmark for rough sketch cleanup. *ACM Trans. Graph. 39*, 6 (2020).