

How to Draw Illustrative Figures?

Yuki Morimoto*
ISIT

Daisaku Arita
ISIT

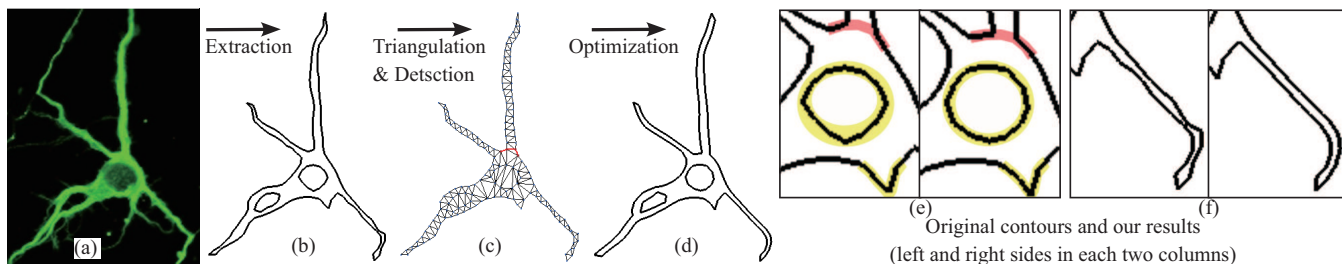


Figure 1: (a) Input image [Millet et al. 07]. (b) Contours extracted from input image. (c) Constrained Delaunay triangulation and detection of collinearity (red). (d) Our result. (e) Comparison of collinearity, roundness and sharpness (yellow). (f) Comparison of parallelism. (left and right sides in each two columns)

1 Introduction

Simplified illustrations are effective for medical purposes in ways: they can quickly convey stories to patients, and avoid unnecessary graphic depiction. Despite their simplicity, such illustrations are not trivial to improvise. In this talk, we propose a new system that automatically transforms medical images into friendly 2D illustrations via an optimization problem.

We observe that many illustrations are simplified or exaggerated: they have smooth lines with partial parallelism, collinearity, and similarity. Taking account these principles of design, we particularly focus on preserving parallelism, collinearity, and similarity of contours, when performing noise removal. Our optimization finds the contours that best match the suggested design principles by solving a nonlinear least-squares problem. As shown in Figure 1, our method is able to enhance some geometric features.

2 Our Method

Feature-Preserving Weighted Smoothing. We resample contour lines in such a way that the points are equally spaced on the original contour lines preserving sharp points. We perform constrained Delaunay triangulation over the contour lines for the discretization and the detection of collinear point indices, e.g. $i, i+1, j, j+1$ by Fu's collinearity criteria [Fu et al. 2011]. Then we perform filtering operations over the curvatures defined in [Eigensatz et al. 2008] and distances at each point. We define the distance as an average of lengths associated with Delaunay edges except for the neighboring points of contours. Our filtering operations diffuse the curvature and the distance value by weighting [Ohtake et al. 2002]. This results in stair-like approximation: the operation enhances the geometric features while removing noises and clustering similar features.

Optimization. We define smoothing energy $E_{sm} = \int_{\Omega} (\kappa - \kappa') ds$, distance energy $E_d = \int_{\Omega} (d - d') ds$, collinearity energy $E_c = \int_{\Omega} (\kappa_c - \kappa'_c) ds$, similarity energy $E_{si} = \int_{\Omega} (\kappa_u - \kappa'_u) ds$, and remaining energies such as position and points intervals of contours E_p, E_{in} defined in the method of Eigensatz et al. [2008], where the contours $\Omega \subseteq \mathbb{R}^1$, ds denotes a line element, κ denotes the curvature of contour points, κ_c denotes the curvature calculated by using collinear point index, κ_u denotes the unsigned curvature value, and primes denote the filtered instances through the configurations

*e-mail: yukim128@gmail.com

of these parameters, respectively. The total energy is defined by $E = k_{sm}E_{sm} + k_dE_d + k_cE_c + k_{si}E_{si} + k_pE_p + k_{in}E_{in}$, where the parameters $k_{sm}, k_d, k_c, k_{si}, k_p, k_{in}$ are the coefficient that are controlled by user's preferences. We employed the Levenburg-Marquardt algorithm to minimize the energy E (See [Eigensatz et al. 2008] for details).

3 Results and Conclusion

We show an example results in Figure 1 generated by solving our minimization problem. We used OpenCV 2.1 to extract the contour lines from a photograph. From the comparison of the red colored edge angles in Figure 1, we are able to see that the collinearity optimization makes these more collinear. The signed curvature optimization removes noise from the contours while enhancing the sharpness and roundness. The distance optimization also improves parallelism.

One of the drawbacks of our method is that finding the parameters for collinearity detection, smoothing, and optimization require careful adjustments. In our experiments, we found that collinearity detection is too sensitive because the applied criterion is not specific for our purpose. Smoothing parameters are also sensitive to the detection of a feature point which is sharp, parallel, or similar in curvature, or unsigned curvature optimization. We expect to semi-automatically assign these parameters by evaluating weighted smoothing or by finding constrains, or by using user interfaces. The sampling interval of the points also affects the optimization sensitively.

In this talk, we proposed an optimization algorithm that improves illustrations using the proposed design principles. We showed that our optimization algorithm increases the quality of input vector images.

References

- EIGENSATZ, M., SUMNER, R. W., AND PAULY, M. 2008. Curvature-domain shape processing. *Comput. Graph. Forum* 27, 2, 241–250.
- FU, H., ZHOU, S., LIU, L., AND MITRA, N. J. 2011. Animated construction of line drawings. *ACM Trans. Graph.* 30, 6, 133.
- MILLET, L. J., STEWART, M. E., SWEEDLER, J. V., NUZZO, R. G., AND GILLETTE, M. U. 2007. Microfluidic devices for culturing primary mammalian neurons at low densities. *Lab Chip* 7, 8, 987–994.
- OHTAKE, Y., BELYAEV, A. G., AND SEIDEL, H.-P. 2002. Mesh smoothing by adaptive and anisotropic gaussian filter applied to mesh normals. In *VMV*, 203–210.