Local Silhouette Rectification For Post-Processing Antialiasing

Alexander Reshetov and Alexei Soupikov

Intel Labs

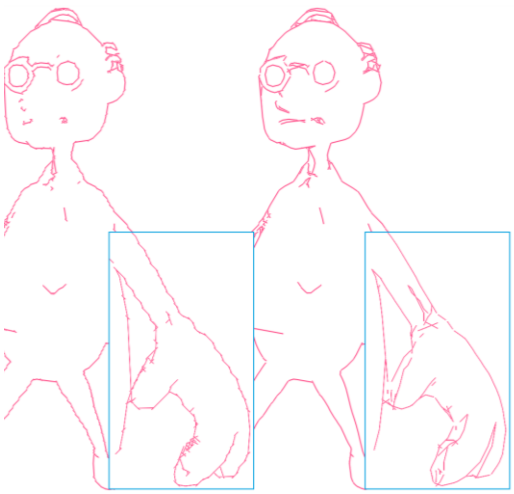

Figure 1. *We improve quality of silhouette lines using only local data.*

Problem

Silhouette detection plays a significant role in different computer graphics disciplines, such as computer vision or super-resolution. Recently proposed post-processing anti-aliasing algorithms [Jimenez et al.] are based on hallucinating silhouettes from sampled data. The quality of the final image depends on the quality of the reconstructed silhouette edges. Another requirement for such algorithms is execution speed, as these algorithms compete with traditional super- and multi-sampling antialiasing. This pretty much excludes global algorithms, such as Fattal's image upsampling via imposed edge statistics [2007], and Kopf & Lischinski's depixelizing approach [2011].

We propose a simplified way to improve silhouette quality that only uses the lengths of the connected linear segments. This enhancement can be used in all image-based algorithms that use silhouette lines and require fast processing.

Local Silhouette Rectification

Fattal [2007] used a sophisticated statistical-based method to infer silhouette edges. We generally use a similar tactic, adopting it for real-time processing. We learn the best way to reconstruct a straight line silhouette and then transfer it to the general case of an arbitrary silhouette. In spirit, it corresponds to the concept of Occam's Razor, assuming the simplest possible explanation for an observed image.

Figure 2 shows a straight silhouette line (dotted red) separating two objects (pink and khaki). When these objects are rasterized, we will get two clusters of pink and khaki pixels separated by green horizontal separation lines. This is the information we have to use to reconstruct an unknown silhouette line. If $\mathbf{y} = 0.5$ intercepts were used, we would get a wave-like silhouette (black line).

Our goal is to express $\mathbf{y}$ as a function of only two variables: ℓ_c and ℓ_o – lengths of the two adjusted separation lines. This way, it will be possible to calculate $\mathbf{y}$ locally, without any additional information.

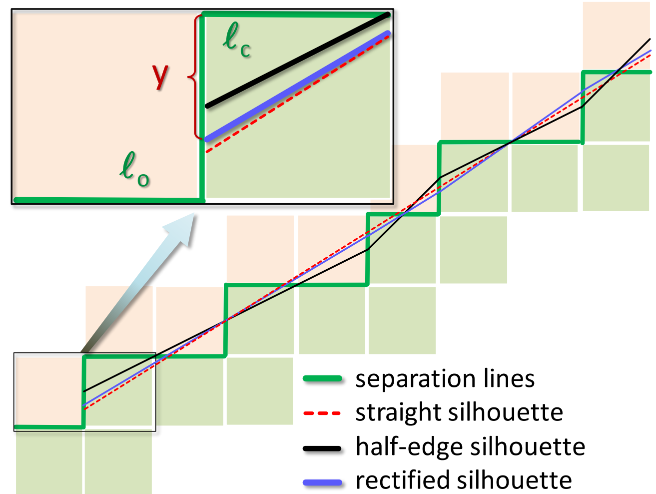

Figure 2. *Fitting a reconstructed silhouette line (blue) to a straight line (dashed red).*

Since we want the silhouette defined by $\mathbf{y}(\ell_c, \ell_o)$ to be as straight as possible, we calculate approximation error by computing absolute slope differences between the reconstructed silhouette segments and the input straight line. These values are averaged for all possible input lines (varied by the slope) and the corresponding segments. We will look for a function, which minimizes this error.

We will formulate the following

> *LSR conjecture*: optimal local $\mathbf{y}$ has a form (q is a constant)
> $\mathbf{y}(\ell_c, \ell_o) = 1/2 + q\,(\ell_c - \ell_o) / (\ell_c + \ell_o)$

In principal, correctness of this conjecture is not required for further derivations. What is important is that this expression allows a significant reduction of the average error, compared with $\mathbf{y} = 0.5$. Still, we believe in correctness of this hypothesis. It is symmetric with respect to ℓ_c and ℓ_o and generally has the right feel.

It might be rather challenging to find an optimal value of q analytically, but it is trivial to do so numerically. We use a family of possible straight lines (parameterized by a slope) and compute an average of ratio $\mathbf{y}_q / \mathbf{y}_{1/2}$. This function has only one minimum at $q = 0.586$, reducing the average error by 3X.

In the general case, we also use the same expression to compute y-intercepts. The only adjustment we make is to clip $\mathbf{y}$ to the [0, 1] interval. If we were to use $q = 0.5$, this would not be required, but $q = 0.5$ has a slightly bigger error. Figure 1 shows results achieved with LSR algorithm.

References

FATTAL, R. 2007. Image Upsampling via Imposed Edge Statistics. In *ACM Trans. on Graphics*, 26(3), Article No.: 95.

JIMENEZ J., AND GUTIERREZ D., AND YANG J., AND RESHETOV A., AND DEMOREUILLE P., AND BERGHOFF T., AND PERTHUIS C., AND MCGUIRE H., AND LOTTES T., AND MALAN H., AND PERSSON E., AND ANDREEV D., AND SOUSA T. 2011. Filtering Approaches for Real-Time Anti-Aliasing. ACM SIGGRAPH Courses.

KOPF, J. AND LISCHINSKI, D. 2011. Depixelizing Pixel Art. In *ACM Trans. on Graphics*, 30(4), 99:1 – 99:8.