Applied Mathematical Sciences, Vol. 4, 2010, no. 14, 667 - 680

Multiscale Flat Norm Signatures for Shapes and Images

Kevin R. Vixie, Keith Clawson, Thomas J. Asaki[1]

Mathematics Department, Washington State University
P.O. Box 643113, 203 Neill Hall
Pullman WA 99164-3113, USA

Gary Sandine[2], **Simon P. Morgan**[2]

Theoretical Division (T-5), MS B284
Los Alamos National Laboratory
Los Alamos, NM 87545, USA

Brandon Price

Department of Mathematics, Walla Walla University
204 S. College Avenue
College Place, WA 99324-1198, USA

Abstract

In this paper we begin to explore the application of the multiscale flat norm introduced in Morgan and Vixie [13] to shape and image analysis. In particular, we look at the use of the multiscale flat norm signature for the identification of shapes. After briefly reviewing the multiscale flat norm, the L^1TV functional and the relation between these two, we introduce multiscale signatures that naturally follow from the multiscale flat norm and its components. A numerical method based on the min-cut, max-flow graph-cut is briefly recalled. We suggest using L^2 minimization, rather than the usual Crofton's formula based approximation, for choosing the required weights. The resulting weights have the dual benefits of being analytically computable and of giving more accurate approximations to the anisotropic TV energy. Finally, we demonstrate the usefulness of the signatures on simple shape classification tasks.

Mathematics Subject Classification: 94A08, 94A12, 49Q15, 28A75,

[1]tasaki@wsu.edu

[2]Supported by the U.S. Department of Energy through the LANL/LDRD program.

26B15

1 Introduction

Since the publication of the paper in 1992 by Rudin, Osher and Fatemi [15], total variation regularization has gained a great deal of attention due to its effectiveness in various image analysis tasks. Let Ω be a domain, often rectangular, on which the graylevel image intensity function or *input* $u : \Omega \to \mathbb{R}$ is defined. The function $f : \Omega \to \mathbb{R}$ will denote the *data*, typically a graylevel image we would like to denoise or restore. The Rudin, Osher and Fatemi (ROF) variational problem,

$$u^* = \text{argmin}_u \, F_{ROF}(u) \tag{1}$$

where

$$F_{ROF}(u) \equiv \int_\Omega |\nabla u| dx + \lambda \int_\Omega |u - f|^2 dx \tag{2}$$

is compellingly simple, yet closely connected to the deep theory that has developed around functions of bounded variation. (A function $u \in L^1(\Omega)$ is said to be a function of bounded variation if $\int_\Omega |\nabla u| dx$, appropriately interpreted, is finite.) The feature that stood out in ROF minimizers was edge preservation – in many cases edge locations were perfectly preserved even while the noise was reduced or removed altogether. Chan and Esedoglu [6] introduced an analogous total variation regularization with L^1 data fidelity:

$$F_{CE}(u) = \int_\Omega |\nabla u| dx + \lambda \int_\Omega |u - f| dx \tag{3}$$

In their beautifully clear paper they establish many useful properties of this closely related functional. Earlier work by Alliney [4] and by Nikolova [14] contained very nice studies of various aspects of the discrete versions of this functional. Because Chan and Esedoglu first introduced the continuous version and were the first to make an in-depth study of its properties, we will refer to this functional as either the L^1TV or the Chan-Esedoḡlu (CE) functional.

Later work by Allard [1, 2, 3] used rather deep tools from geometric measure theory, developed originally to study minimal surface regularity properties for a family of functionals that includes both the ROF and CE functionals. Work by Vixie [16] used simpler notions from geometric measure theory to establish some properties of exact solutions to the CE functional.

In 2007, Morgan and Vixie [13] realized that the CE functional was in fact equivalent to the flat norm from geometric measure theory. This immediately suggested using L^1TV algorithms for the computation of both the flat norm and the flat norm decomposition. It further suggested simple but very useful

generalizations of both the flat norm and the CE functional, both of which would be valuable for work on shapes and images.

In this paper, we will show by example some of the implications of the connection between the flat norm and the CE functional. In particular, we will explore the multiscale flat norm signatures and their use in shape analysis tasks.

2 The Flat Norm and Flat Norm Signature

In this section, χ_E denotes the characteristic function on E, i.e. $\chi_E(x) = 1$ for $x \in E$ and $\chi_E(x) = 0$ for $x \notin E$. In the case that the input u is binary, Chan and Esedoglu observed that the CE functional reduces to

$$F_{CE}(\Sigma, \lambda) = \mathrm{Per}(\Sigma) + \lambda|\Sigma \triangle \Omega|. \tag{4}$$

where Σ is the support of the input $u = \chi_\Sigma$, $\mathrm{Per}(\Sigma)$ is the perimeter of the set Σ, $\triangle$ denotes the symmetric difference, and Ω is the support of the binary data $f = \chi_\Omega$. We will define

$$\Sigma(\lambda) \equiv \arg\min_E F_{CE}(E, \lambda). \tag{5}$$

Note that we are now explicitly tracking the dependence on λ and that we will sometimes use F^Ω_{CE} to indicate the dependence of F_{CE} on Ω.

The *flat norm with scale* λ, $\mathbb{F}_\lambda(T)$ of an oriented 1-dimensional set T is given by

$$\mathbb{F}_\lambda(T) \equiv \min_S \{\mathbf{V_1}(T - \partial S) + \lambda \mathbf{V_2}(S)\} \tag{6}$$

where S varies over 2-dimensional regions, $\mathbf{V_1}$ is 1-dimensional volume (length) and $\mathbf{V_2}$ is 2-dimensional volume (area). We refer to $\{T, S\}$ as the flat norm induced, optimal decomposition. From [13] we have

Theorem 1. *The flat norm of $\partial\Omega$ equals the minimal value of CE functional with data Ω: that is*

$$\mathbb{F}_\lambda(\partial\Omega) = F^\Omega_{CE}(\Sigma(\lambda), \lambda), \tag{7}$$

and

$$\{\partial\Omega, \Sigma(\lambda) \triangle \Omega\} \tag{8}$$

is an optimal decomposition that the flat norm with scale requires.

We note that we have glossed over details such as what we mean by $\partial\Omega$, the boundary of Ω (for nice enough sets it is simply the usual topological boundary) and other fine points. References for this material include Frank Morgan's introduction to geometric measure theory [12] as well as the books

by Evans and Gariepy [8], and Federer [9]. We recommend that you start by looking at Morgan's introduction.

Finally, the three *flat norm signatures* f_e, f_{tmbs} and f_S are defined by

$$f_e(\lambda) = \mathbb{F}_\lambda(\partial\Omega) = F_{CE}(\Sigma(\lambda), \lambda) \tag{9}$$

and the decomposition of $f_e(\lambda)$ into $f_{tmbs}(\lambda)$ and $f_S(\lambda)$ through

$$f_e(\lambda) = f_{tmbs}(\lambda) + \lambda f_S(\lambda) \tag{10}$$

where $f_{tmbs}(\lambda) = \mathbf{V_1}(\partial\Omega - \partial(\Sigma(\lambda) \triangle \Omega)) = \mathrm{Per}(\Sigma(\lambda))$ and $f_S(\lambda) = \mathbf{V_2}(\Sigma(\lambda) \triangle \Omega) = \int_{\Sigma(\lambda)\triangle\Omega} 1dx$.

3 Graph Cuts for CE computation

Kolmogorov and Zabih's paper in 2004 [11] (see also the 2001 paper by Boykov, Veksler, and Zabih [5]) introduced graph cut methods for solving variational problems. Implementations for image analysis functionals are a natural application, see Darbon and Siegle [7] and Goldfarb and Yin [10].

The essential idea is to represent a functional minimization problem as an equivalent minimum graph cut problem. A simple one dimensional example is shown in Figure 1. The image is a characteristic function of eight pixels where 0-1 intensities are indicated by gray-white shading. The graph is constructed by first associating each pixel with a node. Then graph edges are added between nodes whose corresponding image pixels are neighbors in image space, and each edge is given unit weight. Next, we add a source node (s) and a sink node (t) with connecting edges of weight λ to each white or gray pixel node, respectively.

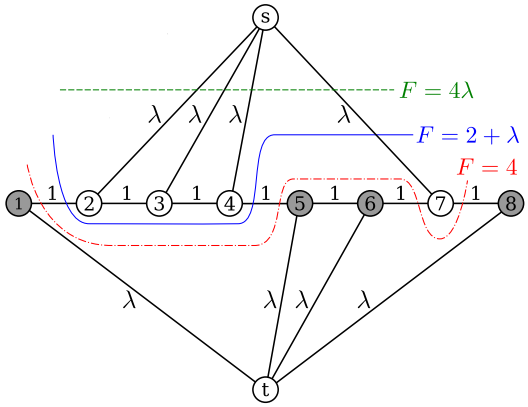

Figure 1: Illustration of a graph cut for a 1-dimensional image.

The capacity of any cut on this graph is equal to the value of $\mathbb{F}_\lambda$ for a particular region S. In the graph, S corresponds to the set of all nodes n for

which either the edge (n, s) or (n, t) is in the cut. For instance, in Figure 1 the solid blue line illustrates a cut for which $S = \{7\}$. Cutting $(s, 7)$ reduces the s-t flow by λ, which corresponds to the $\lambda \mathbf{V_2}(S)$ in Equation 6. Any cut also incurs a penalty equal to the number of times the cutting line passes between image nodes. This penalty is equal to the term $\mathbf{V_1}(T - \partial S)$ in Equation 6. Thus, finding a cut with minimimal capacity is equivalent to computing the flat norm.

It is straightforward to extend this method to two or more dimensions. In our work on characteristic functions represented as binary images on regularly spaced pixels we consider graphs constructed using the 16 neighbor scenario illustrated in Figure 2. The use of nearest (black), next-nearest (red) and fourth-nearest (blue) neighbors yields a better (anisotropic) approximation of the gradient due to the high density of sampling directions relative to simple nearest neighbor computations.

Optimal weights for the graph edges are determined by minimizing gradient computation error on known functions. Let $g_\theta : \mathbb{R}^2 \to \mathbb{R}$ be the linear function whose gradient everywhere is $\nabla g_\theta = (cos\theta, sin\theta)^T$. Also let v_j be the 16 vectors quantifying the image-space neighbor positions. For example, if the image grid is of unit spacing then $v_1 = (0, 1)^T$, $v_5 = (1, 1)^T$, $v_9 = (1, 2)^T$, etc. We choose three edge weights $w^* = (w_1^*, w_2^*, w_3^*)^T$, corresponding to the three types of neighbors in Figure 2, which best approximate ∇g_θ for all θ:

$$w^* = \arg\min_w \int_0^{2\pi} \left(h(w, \theta) - 1\right)^2 d\theta, \tag{11}$$

where

$$\begin{aligned} h(w, \theta) &= \sum_{j=1}^{4} w_1 |\nabla g_\theta \cdot v_j| \\ &+ \sum_{j=5}^{8} w_2 |\nabla g_\theta \cdot v_j| \\ &+ \sum_{j=9}^{16} w_3 |\nabla g_\theta \cdot v_j|. \end{aligned} \tag{12}$$

The three terms in Equation 12 are the nearest, next-nearest and fourth-nearest neighbor terms, respectively. Equation 11 is solved analytically, though the full expression is cumbersome and not reproduced here. We find $w^* \approx (0.1221, 0.0476, 0.0454)^T$. Crofton's formula can be used to suggest 16-neighbor weights of $w^* = (\pi/24, \pi/(24\sqrt{2}), \pi/(48\sqrt{5}))^T \approx (0.1309, 0.0926, 0.0293)^T$. These weights give slightly less accurate approximations of both the magnitude of the gradient $|\nabla u|$ and the *total variation* or *perimeter* term $\int_\Omega |\nabla u| dx$ used in Equation 3.

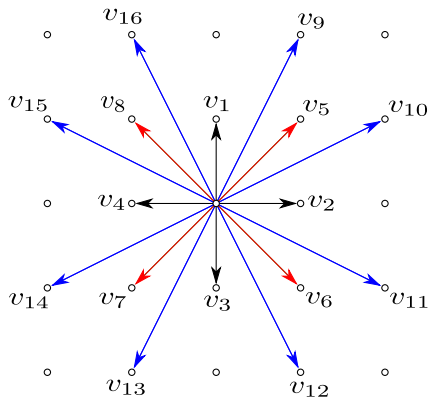

Figure 2: Neigborhood used for 16-neighbor approximation.

We use a graph cut algorithm for CE minimization written by Wotao Yin to calculate minimizers and minimal values from which we extract the signatures and decompositions. The efficiency of the graph cut algorithms was the deciding factor in our choice to use them for the calculation of the CE minimizers.

4 Shapes and shape signatures

We next demonstrate that the flat norm signature can be used as an effective scale-dependent tool in image classification and recognition. For example, suppose we are given a collection of shapes and wish to provide similarity groupings. We will show that (using the multi-scale flat norm) we can extract features of these shapes and, after processing, cluster them appropriately. Similar shapes can then be easily classified based on the clustering of the prior training data.

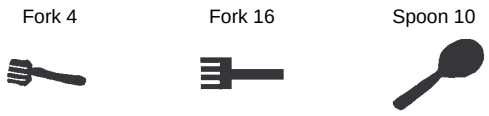

Figure 3: Sample of images to compare. See Appendix for a full list of images used.

We want our classification to be invariant with respect to the scaling of each image, so we normalize our signatures. We would like each image to be scaled so that the area of the shape is 1. If we do not scale the images, signatures of differently sized but otherwise identical images will not match as they should

for our clustering application. We illustrate this in the appendix using the clip art image of fork 4 shown in Figure 3.

Let Φ denote the linear map that scales distances between points by ϕ, so that a scaling of the image Ω is represented by $\Phi\Omega$. Rather than scale the image itself, we can choose a sequence λ^* that will give us a signature $\mathbb{F}_{\lambda^*}(\partial\Phi\Omega)$ that can be scaled to match the signature $\mathbb{F}_\lambda(\partial\Omega)$ of the normalized shape.

Substituting $\partial\Omega = T$ into equation 6, we have

$$\begin{aligned}\mathbb{F}_{\lambda^*}(\partial\Phi\Omega) &= \min_S\{\mathbf{V_1}(\partial\Phi\Omega - \partial\Phi S) + \lambda^*\mathbf{V_2}(\Phi S)\}\\ &= \min_S\{\phi\mathbf{V_1}(\partial\Omega - \partial S) + \lambda^*\phi^2\mathbf{V_2}(S)\}\\ &= \phi\mathbb{F}_\lambda(\partial\Omega),\end{aligned}$$

where $\lambda = \lambda^*\phi$. Thus, letting f_e^*, f_S^*, f_{tmbs}^* denote the signatures of $\Phi\Omega$, the following equations give the normalized signatures:

$$f_e^*(\lambda^*) = \phi f_e(\phi\lambda^*) \tag{13}$$

$$f_S^*(\lambda^*) = \phi^2 f_S(\phi\lambda^*) \tag{14}$$

$$f_{tmbs}^*(\lambda^*) = \phi f_{tmbs}(\phi\lambda^*). \tag{15}$$

We begin with a collection of 24 mixed fork and spoon images (Figure 3). We then calculate the normalized multi-scale flat norm of each image for a sequence of λ values. In Figure 4, we plot signatures for the mass of S. We note that these signatures are always monotonic and that there are abrupt jumps in the signatures. To gain a sense of this, we can examine an image at several selected λ values and see progressively larger radius features absorbed into the mass of S (Figure 5).

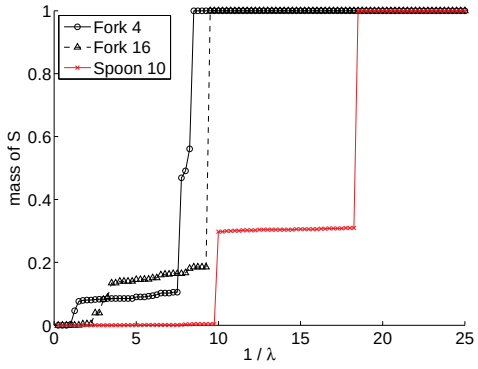

Figure 4: Signatures of S for sample images.

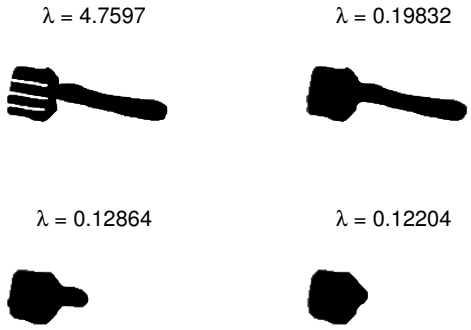

Figure 5: Fork 4 at different λ values.

We can analyze the signature information by graphing the discrete derivative graph or *spectrum* of the mass of S (Figure 6). Here, spikes represent increases in the mass of S. The location of a spike relates to feature scale, while the magnitude of a spike represents how significant that feature is. Spectra with similar spikes (in both location and magnitude) share important feature similarities.

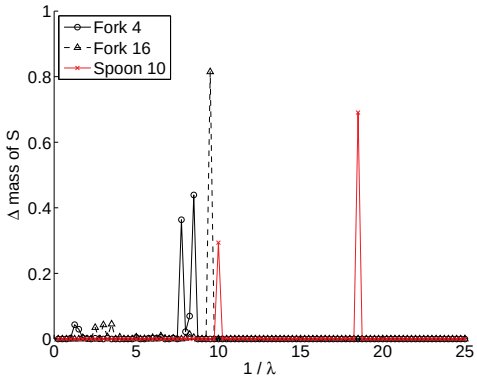

Figure 6: Spectra of S for sample images.

If we attempt to directly cluster this data in a high-dimensional space ($\mathbb{R}^m$ where m is the number of λ values), spikes that do not directly overlap will not be considered close (under a Euclidean metric) regardless of their position along the spectrum. However, it is natural to expect that features at similar scales are significant from an image similarity standpoint. We achieve this similarity by convolving the raw spectra with a Gaussian distribution of width parameter σ equal to the desired similarity scale. The smoothed example spectra are shown in Figure 7. Smoothed spectra of each of seven representative fork and spoon images is shown in Figure 8.

Clustering in $\mathbb{R}^m$ gains more meaning now, but there are two challenges.

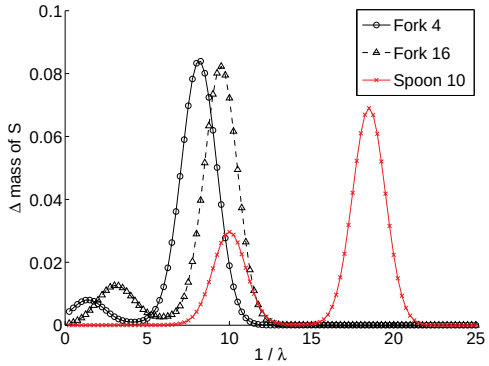

Figure 7: Smoothed spectra of S for sample images. $\sigma = 1$.

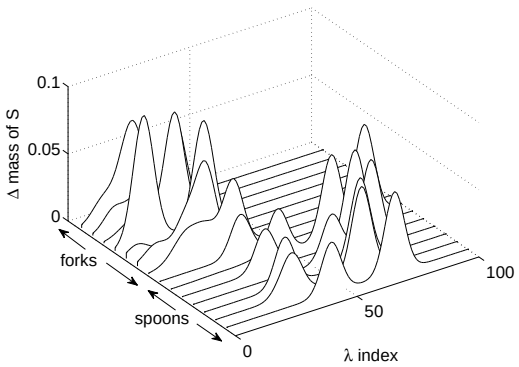

Figure 8: Smoothed spectra for 7 forks and spoons. $\sigma = 1$.

First, many dimensions add no critical information to distinguish between the shapes. Second, it is impossible to visualize the data. We solve both problems by using the singular value decomposition on the matrix of data vectors. Figure 9 shows the data projected onto the subspace spanned by the two most significant singular vectors. Ideally we would see a simple separation of the data into two groups. After this training separation, we can take new images, perform the operations described above, project onto the same subspace, and classify each new image based on the previous grouping. In the present example, sufficiently accurate spoon-fork classification is accomplished using projection onto only the first singular vector.

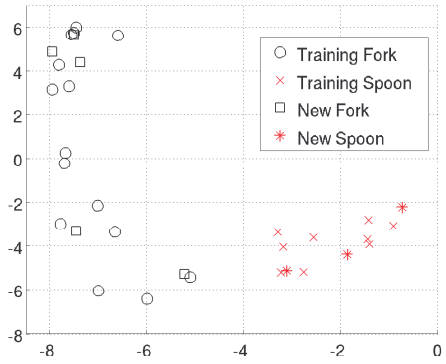

Figure 9: Smoothing $\sigma = 1$. Training fork and spoon data, projected onto the space spanned by the two most significant singular vectors. New fork and spoon data, projected onto the same vectors separate nicely.

As discussed earlier, up to the anisotropic approximation used in the graphcut methods, the multi-scale flat norm is invariant under translations and rotations. For some image processing, this is highly desirable.

A potential problem is caused when shape boundaries exist very near the image sides. This rises from the Neumann boundary conditions used in the graphcut calculations and can easily be fixed by padding the image before calculating the signatures.

5 Acknowledgements and code

Data and code used to generate all results in this paper are available at `http://www.datachallengecooperative.org/code`. We acknowledge and thank Wotao Yin for his code and pointers on its use. The code can be found here: `http://www.caam.rice.edu/~wy1/ParaMaxFlow/`.

References

[1] W. Allard. Total variation regularization for image denoising; I. Geometric Theory. *SIAM Journal on Mathematical Analysis*, 39:1150–1190, 2007.

[2] W. Allard. Total variation regularization for image denoising; II. Examples. *submitted*, 2008.

[3] W. Allard. Total variation regularization for image denoising; III. Examples. *Submitted*, 2008.

[4] S. Alliney. A property of the minimum vectors of a regularizing functional defined by means of the absolute norm. *IEEE Trans. Signal Process.*, 45:913–917, 1997.

[5] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.

[6] T. Chan and S. Esedoḡlu. Aspects of total variation regularized L^1 function approximation. *SIAM J. Appl. Math.*, 65(5):1817–1837, 2005.

[7] J. Darbon and M. Sigelle. A fast and exact algorithm for total variation minimization. In *2nd Iberian Conference on Pattern Recognition and Image Analysis*, volume 3522, pages 351–359. Springer, 2005.

[8] L. Evans and R. Gariepy. *Measure Theory and Fine Properties of Functions*. Studies in Advanced Mathematics. CRC Press, 1992. ISBN 0-8493-7157-0.

[9] H. Federer. *Geometric Measure Theory*. Classics in Mathematics. Springer-Verlag, 1969.

[10] D. Goldfarb and W. Yin. Parametric maximum flow algorithms for fast total variation minimization. Technical report, CAAM TR07-09, Rice University, 2007.

[11] V. Kolmogorov and R. Zabih. What Energy Functions Can Be Minimized via Graph Cuts? *IEEE transactions on pattern analysis and machine intelligence*, pages 147–159, 2004.

[12] F. Morgan. *Geometric Measure Theory: A Beginner's Guide*. Academic Press, 4th edition, 2008.

[13] S. Morgan and K. Vixie. L^1TV computes the flat norm for boundaries. *Abstract and Applied Analysis*, 2007:Article ID 45153, 14 pages, 2007. doi:10.1155/2007/45153.

[14] M. Nikolova. Minimizers of cost-functions involving nonsmooth data-fidelity terms. *SIAM J. Numer. Anal.*, 40:965–994, 2003.

[15] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60(1-4):259–268, November 1992.

[16] K. Vixie. Some properties of minimizers for the Chan-Esedoḡlu L^1TV functional. *arXiv.org*, Oct 2007.

6 Appendix

6.1 Samples used for image classification

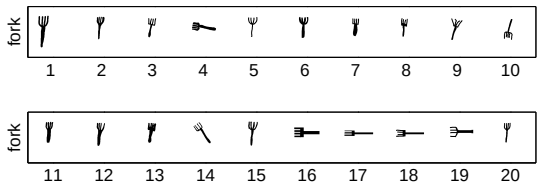

Figure 10: Fork images used in examples. Images 1-15 were used for training, and images 16-20 were tested.

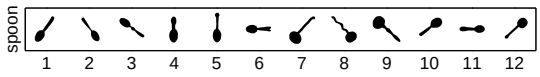

Figure 11: Spoon images used in examples. Images 1-9 were used for training, and images 10-12 were tested.

6.2 Signature normalization example

Figure 12: Fork 4 image for normalization example

Figure 13 shows graphs of the mass of S from the flat norm with scale defined in equation 6 for fork image number 4 and for the same image doubled in size. For our classification application, the signatures of scaled versions of the same image must match.

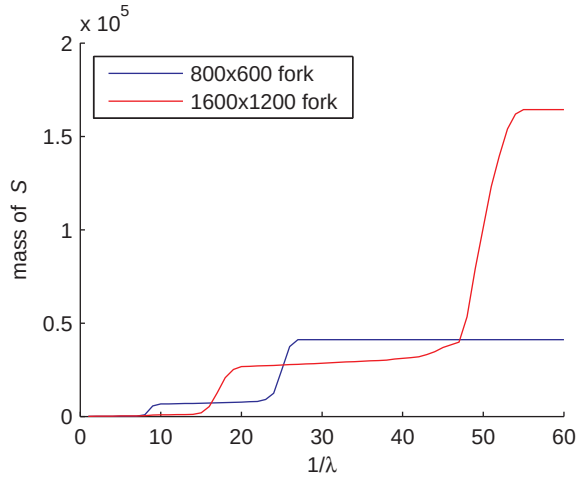

Figure 13: Fork 4 S-signatures for different sized fork 4 images.

Given a sample image Ω of area a, we want the area of $\Phi\Omega$ to be 1, so let $\phi = \sqrt{1/a}$. Figure 14 shows graphs of signatures $\phi^{-2} f_S^*$ from equation 14 which align over the λ^*-axis as they should, and Figure 15 shows normalized signatures f_S^*.

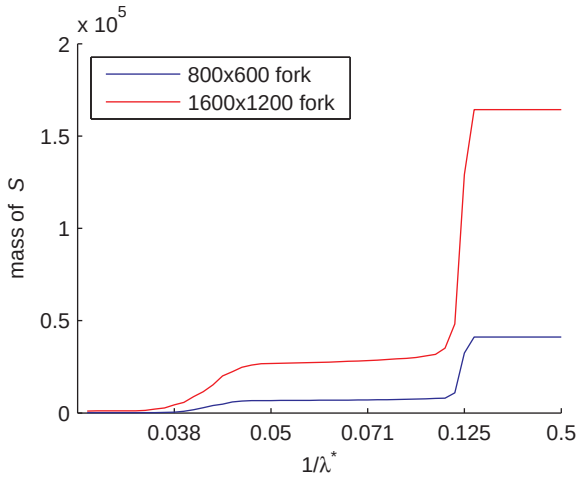

Figure 14: λ-normalized, unscaled fork 4 S-signatures.

Received: June, 2009

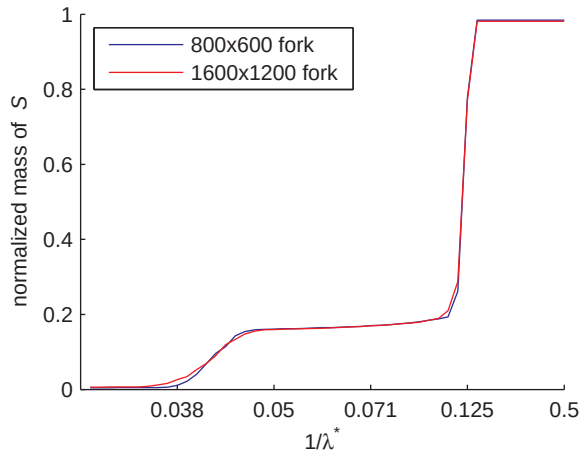

Figure 15: Normalized fork 4 S-signatures.