A Unified Projection Operator for Moving Least Squares Surfaces

T. Ochotta C. Scheidegger J. Schreiner Y. Lima R.M. Kirby Cláudio T. Silva

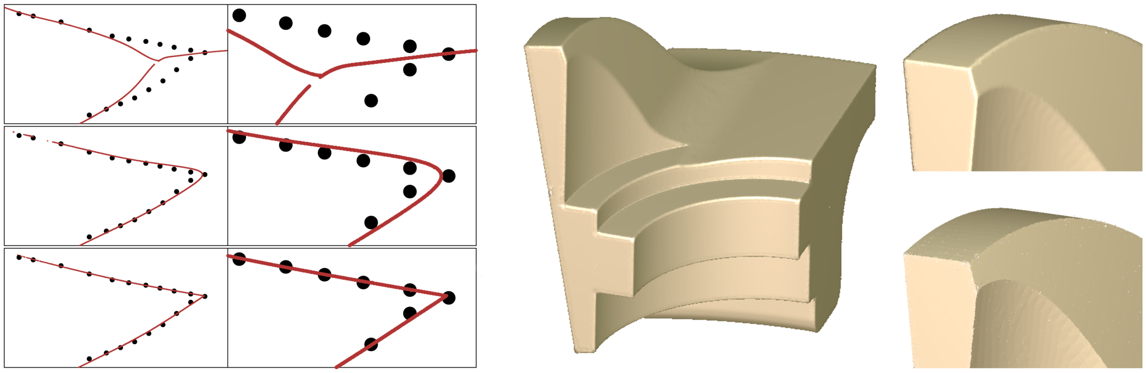


Figure 1: *When Levin's MLS projection is applied to a point set with a sharp angle, the MLS surface can pinch in an undesirable way (left, above). The surface pinches because the optimal local reference frame does not allow any functional approximation to the points, regardless of the density of input points. Our new unified projection operator avoids this artifact and behaves as expected (left, middle). Additionally, it allows the use of different function spaces, which behave as different priors, offering richer reconstruction spaces (bottom-left). On the center, we show a reconstruction of a synthetic model using a quadratic function space. On the top-right corner, the inset shows a closeup of the same model. The bottom-right corner shows the same piece of the model reconstructed with a function space that incorporates C^0 sharp features.*

Abstract

Moving-Least Squares (MLS) surfaces are a popular way to define a smooth surface from a set of unorganized points. In this paper we introduce a *unified MLS projection* consisting of the confluence of several different ideas, each of which independently enhances current MLS strategies or ameliorates MLS deficiencies. We begin by elucidating the shortcomings of the two-phase minimization procedure proposed by Levin [2003]. We show that there are cases intrinsic to the geometry of the underlying surface from which the points are sampled where Levin's projection fails to find an adequate approximation. These shortcomings occur regardless of sampling density or the amount of noise. Our formulation solves this problem by directly fitting a local approximating function to the surface using a unified minimization scheme. Our scheme can be used to create different families of MLS surfaces, depending on the function space used for the fit. This allows specific priors to be used in the approximation, leading to better reconstructions. We present experimental results that show our technique performs well in a wide range of conditions.

CR Categories: I.3.3 [Computer Graphics]: Line and Curve Generation I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, surface, solid, and object representations

Keywords: Moving least squares, point set surfaces, approximating spaces, approximation theory, function space enrichment

1 Introduction

Recently, there has been substantial interest in the area of surface reconstruction from point-sampled data. This work is driven by a set of important applications where the ability to define surfaces out of a set of discrete samples is necessary. For instance, devices capable of acquiring high-resolution 3D models have become affordable and commercially available, and such reconstruction techniques are required for the effective use of these devices. A particularly powerful approach has been the use of the moving least-squares (MLS) technique of Levin [2003] for modeling point set surfaces [Alexa et al. 2001; Amenta and Kil 2004b]. Variants of this framework have become the basis for much of the current point-based modeling work in the graphics, visualization, and computational geometry communities.

The key idea in Levin's formulation is to define the surface in terms of a projection operator, the fixed points of which are the MLS surface. The idea is similar to the seminal work of Lancaster and Salkauskas [1981] for the interpolation and approximation of functions. Levin generalizes this previous work in function approximation theory to accommodate manifolds. Levin's operator involves a non-linear optimization for each point projection, but unlike simpler, subsequent definitions [Adamson and Alexa 2003], it does not require normal information at the points as input. The operator is defined as a two-phase optimization procedure. The first phase computes a reference frame for a local neighborhood of the point being projected with a non-linear weighted least squares fit (Note that this is the main difference from [Lancaster and Salkauskas 1981]). The second (linear) phase finds a best-approximating function in the reference frame computed during the first phase. Typically a tensor-product quadratic is fit to the input points, from which differential-geometric properties of the MLS surface can be approximated. It is possible to skip the second phase, which is equivalent to *fitting* a zero degree function to the point set; this is advocated in many works (*e.g.*, [Amenta and Kil 2004b]). This second phase is critical for our reformulation of the projection procedure. We show that assuming a constant function will make the original MLS definition fail to produce a suitable surface reconstruction for certain geometric configurations, regardless of neighborhood size or sampling density.

There exist cases in which, during the finding of a non-constant function f in the second phase, the reference frame computed by the first phase will generate a poor fit to the input points. The motivating insight for our reformulation lies in the observation that for the same input configuration *there exists another reference frame that allows an accurate fit* (see Figure 3). We address this issue by incorporating the function fitting into the non-linear optimization, thereby unifying the projection procedure into a single fitting phase. Furthermore, we show that it is possible to tailor this new MLS formulation for a variety of geometric processing tasks by changing the function space from which the function f is selected.

This paper makes the following two contributions to the area of point set surface reconstruction: (1) We propose an alternative, unified MLS projection that improves on the work of Levin [2003] in significant ways. In particular, our work corrects and increases the robustness of this seminal technique. Our formulation further develops the theoretical foundations of this approach, helping to elucidate failure points that have existed since the technique's inception over five years ago. Our MLS formulation also provides a conceptually simple way to handle a large class of different priors. We are able to handle sharp features that are not easily handled with other approaches. (2) We show how to implement these techniques in practice. Notably, we describe how to integrate different functional spaces into MLS as a means of capturing sharp features.

2 Related Work

The problem of defining surfaces out of point samples has been actively approached by researchers for many years. Pioneering work has been done by Hoppe et al. [1992] and others [Curless and Levoy 1996; Turk and Levoy 1994; Wheeler et al. 1998] in the context of surface reconstruction, where the primary focus was on building triangle meshes of sampled surface. Early important work in the area was also done in the computational geometry community, particularly in the area of connectivity reconstruction using techniques based on the Delaunay triangulation [Amenta et al. 1998].

During the development of *Point-Based Graphics* as an independent sub-area of interest [Pfister et al. 2000; Rusinkiewicz and Levoy 2000], it was natural to consider the more general problem of defining surfaces directly from point sets. Alexa et al. [2001]'s Point Set Surfaces was one of the first papers in this area. It showed how a simple and effective representation could be achieved by the use of a moving least-squares (MLS) technique [Levin 2003], even in the case of noisy sets of points. Other related formulations followed, e.g., [Amenta and Kil 2004b; Fleishman et al. 2003; Pauly et al. 2003; Xie et al. 2003].

At this point, there are many variations and extensions of the initial MLS approach to defining point set surfaces. A set of popular techniques are based on defining the point projection in terms of a combination of weighted centroids and a normal field [Alexa and Adamson 2007; Alexa and Adamson 2004; Adamson and Alexa 2003; Amenta and Kil 2004b]. This type of projection defines the surface in terms of a level set of a scalar function. This function can be evaluated very efficiently, but an iterative method is required to project points onto the surface. Also, their simplicity makes them more suitable for analysis that give strong theoretical guarantees [Kolluri 2005; Alexa and Adamson 2007; Dey and Sun 2005; Amenta and Kil 2004b; Bremer and Hart 2005]. We note that some of these linear techniques require normal information, which may be unavailable or unreliable. Guennebaud and Gross [2007] propose an algebraic framework for defining MLS projections; their work includes a more robust technique for normal estimation.

There is also continued interest in further analysis of the two-phase approach using Levin's MLS [Levin 2003] which involve a nonlinear projection, e.g., [Lipman et al. 2006; Fleishman et al. 2005; Gal et al. 2007]. The work of Amenta and Kil [2004a] raised many important practical and theoretical issues, including what happens to the projection near edges and corners; they show that the original projection sometimes has undesirable behavior in those locations. Fleishman et al. [2005] propose to use robust statistics and a modified projection scheme for recovering sharp features as well as increasing stability of the projection operator near features. This approach is sound and produces good results, but it introduces extra processing steps which break the natural elegance of Levin's formulation. Their work can be seen as introducing the use of priors in the reconstruction. An alternative formulation is done in [Gal et al. 2007].

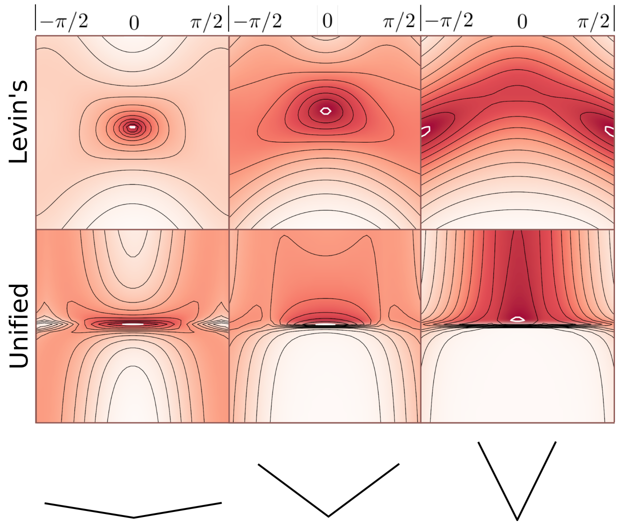


Figure 2: *Flood contours of the energy function for several input point configurations are shown for Levin's projection $\mathcal{P}_L$ (Equation (1), top row), and our unified approach $\mathcal{P}_U$ (Equation (6), bottom row). The horizontal and vertical axis correspond to a parameterization of all possible reference frames, whose normal is at angle θ from $(0,1)$, and a distance ρ from the center point of the wedge, respectively. Notice that for the sharpest wedge, the minimum (always encircled in a white contour) of the energy function for $\mathcal{P}_L$ appears at $\theta = \pi/2$ for Levin's formulation, while the minimum for $\mathcal{P}_U$ is at $\theta = 0$ in all situations.*

3 Mathematical Formulation

The final mathematical result of this paper, which we call a *unified MLS projection*, consists of several different ideas, each of which independently enhances current MLS strategies or ameliorates MLS deficiencies. We split our discussion into the following five steps. To create a consistent nomenclature for our discussion, we begin by reviewing Levin's (two-phase) MLS projection operator [Levin 2003]. Secondly, we identify a deficiency in the two-phase strategy: it misses the appropriate reference frame for certain point set configurations. We introduce the concept of a single-phase MLS projection operator with closest-point projection as a solution to the aforementioned problem. Thirdly, we employ the idea of enriched MLS spaces, which when combined with closest-point projection operators, can capture local regions of non-smoothness without compromising the smoothness of the rest of the surface. Fourthly, we present our unified MLS operator – an operator which combines the idea of a single-stage minimization process and the use of enriched spaces with closest-point projection. The new unified operator finds the appropriate reference frame in situations where Levin's projection fails and allows sharp feature reconstruction. As our final step, we prove that the new operator is indeed a projection.

3.1 Levin's Projection

Levin's projection [Levin 2003], $\mathcal{P}_L$, consists of two stages: finding a local reference frame for the surface analogous to the traditional differential-geometric reference frame and selecting a function over that frame which approximates the data. To be more specific, in the first stage, a local reference frame $H_{r,q}$ consisting of the plane intersecting q and having a normal $(q-r)/|q-r|$ must be determined. Let the orthogonal projection onto this plane be denoted by $h_{r,q} : \mathbb{R}^3 \rightarrow H_{r,q}$. For a given point r, and input points p_i, the plane $H_{r,q}$ is determined by finding q which satisfies the following non-linear optimization problem:

$$\arg\min_{q} \frac{\sum_i |h_{r,q}(p_i) - p_i|^2 \omega(|p_i - q|)}{\sum_i \omega(|p_i - q|)}, \qquad (1)$$

where ω is a strictly positive, smooth weighting function (often a Gaussian) whose argument contains q. In practice, the minimum is often found using a Powell method [Press et al. 1992, Chapter 10.5]. Once this local reference frame has been found, the second phase is performed in which a function $f \in \mathcal{F}, f : H_{r,q} \to \mathbb{R}$ is constructed to fit the data points p_i in the weighted least-squares sense. Let $g_{r,q} : (H_{r,q} \times \mathbb{R}) \to \mathbb{R}^3$ be the linear function mapping the plane and a scalar value back into the global $\mathbb{R}^3$ domain (similar to h^{-1}). With r, q subscripts omitted for clarity, the function f can then be found through the following minimization:

$$\arg\min_{f} \frac{\sum_i |g(h(p_i), f(h(p_i))) - p_i|^2 \omega(|q - p_i|)}{\sum_i \omega(|q - p_i|)}. \qquad (2)$$

For a fixed value of q, the weights are constant. This combined with $\mathcal{F}$ being a finite-dimensional linear function space allows for the function f to be computed directly by solving the resulting linear system. After the function f is found, Levin's projection of r onto the MLS surface is defined as $\mathcal{P}_L(r) = g(h(r), f(h(r)))$, *i.e.* the projection of r onto f in the direction $(q - r)/|q - r|$.

3.2 Single-Phase Minimization with Closest-Point Projection

The two-phase formulation works well when applied to most, but not all, point configurations. The breakdown of the two-phase procedure comes as a consequence of the explicit dependency of the second phase on the results of the first phase. Even if the function space used in the second phase of Levin's procedure were capable of accurately representing the local surface, it requires a reasonable reference frame over which to construct the functional approximation. We know from differential geometry that an appropriate plane must exist: the question is whether the first phase of $\mathcal{P}_L$ can find it.

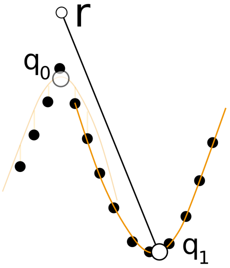
Unfortunately, the first phase of the process fails to produce the expected local reference frame (in the differential-geometric sense) when the points p_i are in certain, but common, geometric configurations. As an example, consider a wedge that is sampled noiselessly with angle ϕ between the supporting lines which meet at $(0, 0)$. Examining the result of the first phase of $\mathcal{P}_L(0, 0)$ as a function of decreasing angle, the (unique) minimum of the energy function will switch at $\phi \leq \pi/2$ from a reference frame which intersects the two supporting lines away from $(0, 0)$ to a reference frame congruent to the bisector of the wedge. This reference frame is inadequate for accomplishing surface fitting: the two supporting lines will be mapped to the same subset of the domain, so no function will correctly approximate it. The consequence of this deficiency is demonstrated in Figure 1(b), where the MLS projection operator has pinches and extends away from the points p_i.

This example demonstrates that the greedy approach of decomposing the global optimization into two phases – the first phase being fundamentally limited to planar approximations and the second phase consisting of a richer approximation space – fails to create satisfactory approximations of the surface. Furthermore, enhancement of the functions used in the second phase cannot, in general, correct (or adequately compensate for) the problems generated by the first phase. A similar issue was identified by Amenta and Kil [2004a]; they proposed an integral-based formulation of the error function. Their formulation solves all problems produced by inadequate sampling of the surface, but for geometric configurations that are intrinsically bad, the integral-based formulation still produces non-manifold surfaces.

Instead of applying the greedy approach to this problem as done by Levin's formulation, we propose that the finding of a surface approximation should be done as single phase where the reference frame is chosen so that it will *minimize the error of the best function space approximation in that reference frame*, instead of the error induced by the distance between the points and the plane. Since $g(h(p_i), f(h(p_i))) = h(p_i)$, we can then combine Equations (1) and (2) into a single-phase non-linear optimization problem given by:

$$\arg\min_{q,f} \frac{\sum_i |g(h(p_i), f(h(p_i))) - p_i|^2 \omega(|q - p_i|)}{\sum_i \omega(|q - p_i|)}. \qquad (3)$$

Let the operator $\mathcal{P}_S$ based upon this optimization process be defined as $\mathcal{P}_S(r) = g(h(r), f(h(r)))$. Note that in the case where $\mathcal{F} = \{x \mapsto 0\}$, Equation (3) is equivalent to Equation (1), and hence $\mathcal{P}_S$ is equivalent to $\mathcal{P}_L$. However, when $\mathcal{F}$ is non-trivial the minima of Equation (3) will disagree with the minima of Equation (1). The resulting projection operators will then be different. In many situations, this modification ameliorates the deficiencies with $\mathcal{P}_L$ since the error landscapes look as shown in Figure 2. A tacit advantage of the two-phase approach, however, is that it provides regularization by constraining the approximation. The new minimization process within $\mathcal{P}_S$ is not sufficiently constrained. With q and f both changing simultaneously, the operator is now free to find minima anywhere in the surface. The situation in the inset figure illustrates the issue – suppose we are attempting to approximate a surface with $\mathcal{F}$ chosen as the set of all possible parabolas. Even though there exists a reasonable fit to the surface close to r using q_0, there is an even better fit further away from it using q_1, *with a continuous, monotonically decreasing path in the parameter space existing between the two points*. Hence the optimization procedure naturally proceeds down the path and finds the best parabolic fit. Without the addition of further constraints, the new procedure is not very useful as a surface reconstruction technique since there will not be any points on the MLS surface that are close to r.



The constraint absent in $\mathcal{P}_S$ which is present in $\mathcal{P}_L$ is that projections onto the reference frame $H_{r,q}$ *are naturally closest-point*. This forces $\mathcal{P}_L(r)$ to be close to r during the first phase of the minimization, preventing optimizers from "running away" towards globally good solutions. Based upon this observation, we redefine $\mathcal{P}_S$ as resulting from the following non-linear optimization problem:

$$\arg\min_{q,f} \frac{\sum_i |C_f(p_i) - p_i|^2 \omega(|C_f(r) - p_i|)}{\sum_i \omega(|C_f(r) - p_i|)}, \qquad (4)$$

where $C_f(x)$ is the closest point projection of x onto the function f. This modification still has the property that when $\mathcal{F} = \{x \mapsto 0\}$, $\mathcal{P}_S = \mathcal{P}_L$, but is different otherwise.

3.3 Enriched MLS with Closest-Point Projection

Independent of the aforementioned issue, a second well-known deficiency in Levin's projection procedure is its inability to capture sharp features. As has been identified by others, this deficiency fundamentally due to the choice of function space $\mathcal{F}$ from with

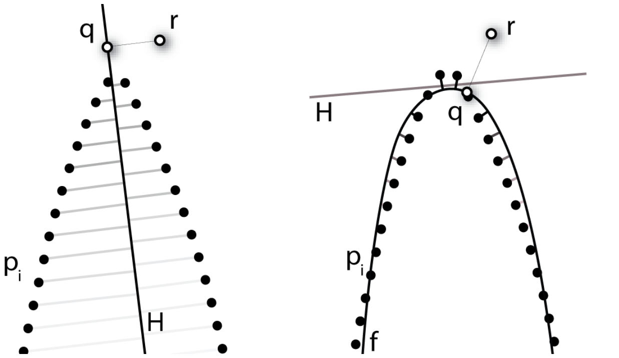


Figure 3: *When projecting a point r onto the surface, $\mathcal{P}_L$ (left) may find a reference frame H that does not naturally allow the input points to be approximated by a function. Our approach (right) searches for H and the approximating function f simultaneously. Residuals are measured differently. In $\mathcal{P}_L$, the error residuals are measured perpendicular to H, and weighted by the distance to q. In $\mathcal{P}_U$, the residuals are measured by closest point distances, weighted by the closest point projection of r onto f.*

which one attempts to approximate the surface. We propose to augment the classic polynomial space used for $\mathcal{F}$ by functions which have the features we seek within our reconstruction. Our goal to introduce a projection procedure that allows different families of functions to be used; this enriched MLS projection allows reconstruction of C^0 features directly from the function space.

We define $\mathcal{P}_E$ as being the same two-phase process as with $\mathcal{P}_L$ with $\mathcal{F}$ now chosen to contain both smooth functions (*e.g.* polynomials) and non-smooth functions (*e.g.* finite element hat functions). The Levin procedure would work with this minor adjustment, but without modification might produce surfaces which are everywhere continuous but nowhere differentiable. To avoid this problem, we enforce that the projection to be used in the second phase will be a closest-point projection onto the local surface approximation.

To recapitulate, the first phase of $\mathcal{P}_E$ is identical to that of $\mathcal{P}_L$: the minimization of Equation (1). We replace Equation (2) by

$$\arg\min_f \frac{\sum_i |C_f(p_i) - p_i|^2 \omega(|C_f(r) - p_i|)}{\sum_i \omega(|C_f(r) - p_i|)}, \tag{5}$$

where $C_f(x)$ is the closest point projection of x onto the function f, and the projection is defined as $\mathcal{P}_E = C_f(r)$. We note that although $\mathcal{P}_E$ ensures the projected points are locally closest to the approximating surface, it suffers from the same reference plane problem as $\mathcal{P}_L$.

The enrichment of $\mathcal{F}$ has the consequence that $\mathcal{P}_E$ is capable of naturally reconstructing sharp features if those features are present on $\mathcal{F}$. This happens because near a point with discontinuous derivatives, there will be a k-dimensional neighborhood ($k > 1$) of points B for which $\forall b \in B, C_f(b) = x$. In other words, all points in B project to the singularity. Since the weights in Equation (5) are determined only by $C_f(r)$ and the input points p_i, if a function is a minimum for one point b in B, it will be one for all of them. This means that the singularity will be *exactly* reconstructed: the surface is only C^0 on $C(b)$. Figure 4 illustrates the argument.

3.4 The Unified MLS Operator

We propose a unified MLS operator which combines the single-phase characteristics of $\mathcal{P}_S$ with the enrichment ideas of $\mathcal{P}_E$. We define the unified operator as the solution of the following non-linear optimization problem:

$$\arg\min_{q,f} \frac{\sum_i |C_{H,f}(p_i) - p_i|^2 \omega(|C_{H,f}(r) - p_i|)}{\sum_i \omega(|C_{H,f}(r) - p_i|)}, \tag{6}$$

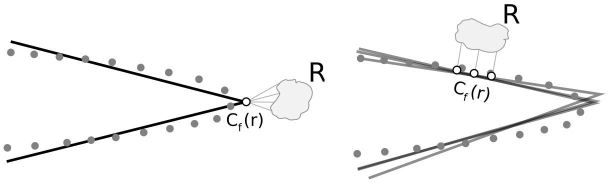


Figure 4: *$\mathcal{P}_U$ can reconstruct C^0 surfaces when the function space $\mathcal{F}$ contains non-smooth functions (left), but still produces smooth surfaces away from the singularities (right). See Sections 3.3 and 3.4.*

where $C_{H,f}(x)$ is the closest point projection of x onto the function f defined over the reference frame H (or simply $C_f(x)$). Then, $\mathcal{P}_U(r) = C_{H,f}(r)$. This combines the advantages of both approaches. The closest point projection onto the local approximation ensures that $\mathcal{P}_U(r)$ will never drift too far from r.

Additionally, by incorporating richer function spaces into the single-phase optimization, $\mathcal{P}_U$ can *exactly reconstruct* functions from $\mathcal{F}$ in the case of compactly supported weighting kernels (this is a feature shared with the tagging-based system of Reuter et al. [2005]). By this we mean that if there is a piece of the surface for which the point set represents some function $f \in \mathcal{F}$ in some reference frame H, such that any point that fails to fit f exactly has zero weight, $\mathcal{P}_U$ will always project to a point with the same q and f, namely the one for which the residuals are zero. Thus $\mathcal{P}_U$ reconstructs functions from $\mathcal{F}$, even in the case of singularities. An important consequence is that in the case of noiseless sampling, it is theoretically possible to reproduce a set of desired surface features from the acquired data by creating $\mathcal{F}$ as linear combinations of basis functions that correspond to the features.

In the case where $\mathcal{F}$ contains non-smooth functions, a d-dimensional ball R might collapse to a point $C_f(r)$ at C^0 features under the closest point projection (see Figure 4). Since this is the point that determines the weights, if $\mathcal{P}_U(r) = C_f(r)$ for some $r \in R$, then for any two points $r_1, r_2 \in R$, $\mathcal{P}_U(r_1) = \mathcal{P}_U(r_2)$. Hence the surface retains the C^0 feature. The careful reader might be concerned about the singularities in $\mathcal{P}_U$: if it generates a singularity in a particular point on the MLS surface, will it not generate singularities everywhere? The answer is no. If $C_f(r)$ projects to a smooth region of f, then small changes in r produce small changes in $\mathcal{P}_U(r)$, and the surface is still smooth.

3.5 Proof of Projection Property

An important question that must be answered is whether or not our new operator, $\mathcal{P}_U$, is indeed a projection. If this were not the case, the point set generated by $\mathcal{P}_U$ could not possibly be a $2-$manifold embedded in R^3, which is a primary goal of surface reconstruction. We start by showing an alternative presentation of the proof that $\mathcal{P}_L$ is a projection. We refer the reader to Step 1 of Section 3 in [Levin 2003]. Assume Levin's operator was applied to a point r such that $\mathcal{P}_L(r) = q$, and $r \neq q$. We will now examine the effect of $\mathcal{P}_L$ in a one-dimensional neighborhood of r along $(q-r)/|(q-r)|$ (Levin calls this direction $a = a(q)$). Let $r' = r + u(r-q), u \in (-\epsilon, \epsilon)$, with ϵ such that the minimum in Equation (1) is unique. It is clear that $a' = (q-r')/|q-r'| = a$, so using q and a as directions for projecting r' is admissible. Then, the first and third conditions in [Levin 2003] hold for r', and the second is satisfied by construction. The consequence is that q is the result of the first stage of Levin's projection for all u. Since the second stage of the projection procedure only moves q in the direction a and does not depend on r, the final result is

$$\nabla_a(P_L(r)) = \mathbf{0}. \tag{7}$$

This implies that the null space of the Jacobian $J(P_L(r))$ is non-trivial, and the Jacobian is rank-deficient. From this observation it follows that P_L is a projection.

In order to prove that our operator is indeed a projection we will similarly show that the Jacobian of $\mathcal{P}_U(r)$ is rank-deficient. We use notation as used previously: $\mathcal{P}_U(r) = q, r \neq q, a = (q-r)/|(q-r)|$, and so forth. If q minimizes Equation (6) for r, then there exists an ϵ for which any $r' = r + u(r-q)$, $u \in (-\epsilon, \epsilon)$ has the same minimizer. First note that $C_{H,f}(r') = q$ for all u, and therefore $\nabla_a(C_{H,f}(r)) = \mathbf{0}$. This is because the closest point projection follows a distance field whose gradient agrees with a. Now, since the only dependence of Equation (6) on r is through $C_{H,f}(r)$, applying the chain rule results in the following:

$$\nabla_a(\mathcal{P}_U(r)) = \mathbf{0}. \tag{8}$$

Again, since the null space of the Jacobian is non-trivial, $\mathcal{P}_U$ is a projection. In the case of $r = q$, we simply set a to be in the direction of the normal of f at q, and the same argument applies. This result is quite general, as any function space used in $\mathcal{P}_U$ will produce a projection operator.

4 Implementation

Implementing the unified projection operator proposed above is only slightly more involved than the two-phase Levin MLS projection. In particular, the significant differences in the implementation are how residuals are computed and how we actually specify the function spaces. In this section, we begin by describing the details of our minimization procedure. We then present a discussion of how to efficiently implement closest-point projections – first for polynomial spaces and then for generalized (enriched) spaces.

4.1 The Optimization Procedure

In our implementation, we use Powell-set optimization, which does not rely on derivative information but still offers quadratic convergence in parabolic regions [Press et al. 1992]. For the actual optimization, we typically have $n+3$ parameters involved, where n is the dimension of $\mathcal{F}$. The three extra parameters determine the center q of the coordinate frame H. From this information we can deduce the normal, which is given by $(r-q)/|r-q|$. A large class of function spaces is closed under affine transformations of the coordinate frame (in other words, $f(x) \in \mathcal{F}$ implies that there always exist a $g(x) \in \mathcal{F}$ that is the composition of f with an affine mapping). However, sometimes it is desirable to use function spaces where this is not the case (for example, $F(x) = \{f : f(x) = k|x|\}$ is not closed under non-uniform scaling of the coordinate axes). In order for $\mathcal{P}_U$ to be invariant under such transformation, we explicitly introduce the terms for M in the optimization.

As initial estimates for $\mathcal{P}_U$ in the case of polynomial function spaces, we use the values of q and f from $\mathcal{P}_L$. For the same reasons outlined in Section 3.2, we use an additional initial estimate of a point whose plane goes through q but whose normal is rotated $\pi/2$ around the minor eigenvector of the neighborhood's covariance matrix. The initial f in this case is a linear least-squares fit. For the sharp feature reconstruction, we use the same two initial estimates for q, but pick the direction vectors to evenly spread around the normal, with zero latitude. Even though the error landscape of $\mathcal{P}_U$ is considerably more complicated than that of $\mathcal{P}_L$, we have found that when using the initial estimates as described above, the Powell set optimizer does not have significant problems with the optimization.

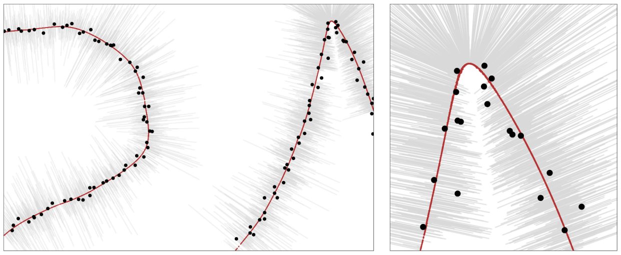


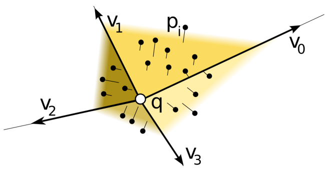
Figure 5: *Like Levin's MLS surface, our unified projection is quite resilient to noisy and irregularly sampled inputs. With a smooth function space, $\mathcal{P}_U$ has the extra benefit of properly rounding sharp corners (right). The grey lines show the paths of the point projections.*

4.2 Closest-point projections

Our definition of $\mathcal{P}_U$ allows different function spaces to produce very different surfaces. Each different f in the function space F is used by $\mathcal{P}_U$ in essentially a single way: points p in R^3 are projected to their closest point $C_f(p)$. The error is then evaluated as $|p - C_f(p)|^2$. In what follows, we describe how we implemented closest-point projection in the spaces we present.

Closest-Point Projection Onto Polynomials. The simplest function spaces we use are P_d^n, the space of all polynomials of total degree n in d dimensions. In particular, we have implemented the projection procedure for P_2^2 and P_3^2. Notice that because the center of the coordinate system q is part of the optimization, the procedure automatically closes the space of functions over transformations of the form $g(x) = f(x+k)$ and $g(x) = f(x) + k$. In the case of polynomials, this means we can drop terms of total degree 1 or less. In the two dimensional case, we use the function space $F(x) = \{f(x) = kx^2\}$, and find the closest point by solving the resulting cubic equation in closed form. In the three-dimensional case, we use $F(x,y) = \{f(x,y) = ax^2 + bx^2\}$. This causes F to not be closed under rotations of the coordinate frame. However, we restore the property by using the technique described in Section 4.1 and introducing an extra parameter to the optimization that rotates the coordinate system. By forcing any cross terms on P_3^2 to be zero, projecting a point onto the closest point is reduced to finding roots for a fifth-order polynomial, which we solve numerically by performing a QR reduction on the companion matrix.

Alternative function spaces. One non-conventional function space we use is capable of reconstructing sharp line-like features that might join, n at a time. This achieves a result similar to what Fleishman et al. [2005] have presented, but without the need for sophisticated statistical machinery.



We define a function $f \in F$ by a set of n unit vectors, $v_0, \ldots, v_{n-1}$ that wind around $(0,0)$ consistently. This is easily described by n pairs of latitude-longitude values, constrained such that the longitude values are ordered. The function is then defined as the set of points created by non-negative combinations of consecutive vectors. Equivalently, the function is the set of infinite triangles with center $(0,0)$ and edges supported by consecutive vectors. In practice, we lift the ordering constraint and explicitly sort the vectors. This makes the optimization algorithm much simpler, and the theoretical disadvantage of making the error landscape discontinuous has not been found to be an issue in practice. Closest-point projection on this space is not only trivial to implement but is also very

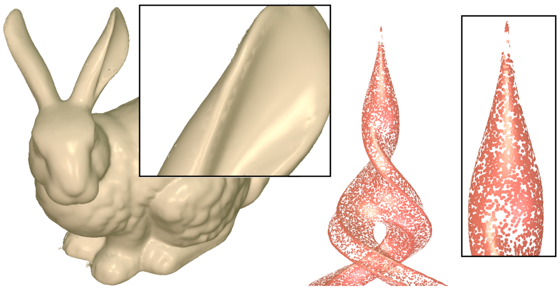


Figure 6: *Reconstructions of the Stanford bunny and the Twirl model using $\mathcal{P}_U$ and quadratic function spaces.*

efficient: we simply perform projections on the infinite triangles and pick the closest point (projecting a point on an infinite triangle is just projecting on a regular triangle but ignoring one of the support lines). In our results, we use $n = 4$. Notice that in this case, if bases $\{v_0, v_1, v_2\}$ and $\{v_2, v_3, v_0\}$ are not full-rank, then the function f will be an edge where the supporting planes are the spans of the two sets.

A general algorithm for performing closest point projection on a function space is also possible when information about the function's fundamental forms is available. With the first fundamental form we can evaluate the surface normal, and so we can define closest point projection as one where the direction of the projection agrees with the normal of the surface. If the second fundamental form is available, we can use more sophisticated algorithms that use derivative information, such as BFGS pseudo-Newton methods [Press et al. 1992].

5 Experiments

Our proposed operator has several desirable features, compared to the original MLS definition and subsequent work. It directly fits local functions to the input, and so is less susceptible to incorrect minima arising from an inadequate reference frame. Therefore, some features from the sampled model might be "pinched" in these definitions. In our unified projection, these corners are "rounded" — the behavior is more similar to a surface under low-pass filtering. This can be seen in Figures 1, 5, and 6.

At the same time, our formulation can also be used to reconstruct C^0 features of the surface. If the functions we are using for the optimization are only C^0, there will be 3-dimensional balls in R^3 that project down to a single point, as illustrated in Figure 4. This runs counter to the intuition that our surfaces should be as continuous as the weighting functions, as Levin conjectures for $\mathcal{P}_L$ in [Levin 2003]. To the best of our knowledge, this is the first technique to directly incorporate sharp features, without formulations that use intersections of smooth surfaces or explicit tagging [Fleishman et al. 2005; Reuter et al. 2005]. It is also possible to reconstruct sharp features from 3D data. Figure 7 shows a reconstruction of the fandisk and a CSG model using a function space that contains nonsmooth functions. Notice that the reconstruction is smooth away from the singularities.

The flexibility afforded by $\mathcal{P}_U$ can be exploited for unusual applications. One can define the function space $\mathcal{F}$ used by the unified operator to contain the expected local surfaces for the model being reconstructed. By changing the function space in this way, we can interpret this unified formulation in terms of providing *surface priors* for the reconstruction. One interesting application of this technique is for automatic completion. In Figure 8, we show two examples of reconstructions from an incomplete set of input points. On the left, the function space $\mathcal{F}$ only contains smooth functions, so a smooth surface is reconstructed. In contrast, the right column shows a case where $\mathcal{F}$ includes functions with sharp features, so a sharp point is reconstructed to fill the gap.

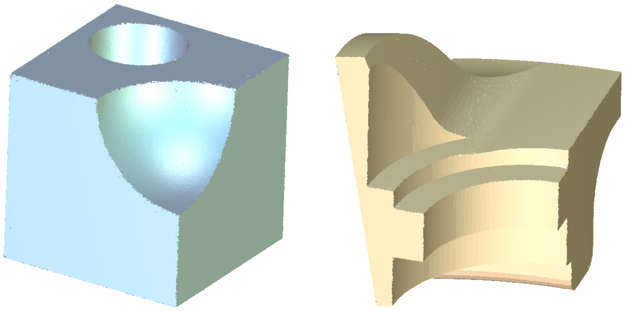


Figure 7: *The 3D function space described in Section 4.2 can be used to reconstruct models with sharp features. On the left, our reconstruction of a synthetic CSG model. On the right, our reconstruction of the fandisk model.*

Performance. Our prototype C++ implementation of $\mathcal{P}_U$ performs comparably to that of Fleishman et al. [2005]. Besides the techniques in Section 4.1 used to make closest-point projection efficient, our only other fundamental optimization was to run projections in parallel, taking advantage of modern multicore workstations. We believe that using GPUs would lead to large increases in performance, as reported for example in [Guennebaud and Gross 2007], and believe this is an important avenue for further investigation. Using the function space P_2^3 in areas of large curvature, the optimizer needs to perform more work, and our $\mathcal{P}_U$ is between one and two orders of magnitude slower than Levin's $\mathcal{P}_L$. The function space consisting of planar wedges is significantly faster, performing around five times faster than the quadratic function space. Our memory usage is essentially the same as for other MLS implementations. Aside from the point set itself, the only data structure kept in memory is a kd-tree used to perform k-nearest neighbor searches and range queries.

6 Discussion

It is interesting to consider the conditions that cause the breakdown of the original MLS operator. This happens when the minor eigenvector of the covariance matrix switches from being aligned with the normal of the expected reference frame to being tangent to the frame. We believe that this is related to the uniqueness of the minor eigenvector of the covariance matrix and how it changes over the surface. Bremer and Hart [2005] have recently used a related technique for proving sampling conditions under which an MLS formulation properly reconstructs a surface. One can see the breakdown of $\mathcal{P}_L$ as a contrapositive analog of their result.

There have been many proposals to accommodate sharp features in surface reconstruction using MLS. The work on enriched, reproducing kernel Hilbert spaces by Reuter et al. [2005] requires explicit tagging of sharp features. Fleishman et al. [2005] have proposed a statistical framework that automatically segments neighborhoods into smooth subsets. While this makes it possible to reconstruct sharp curves, the scheme only works if the C^0 feature is reconstructible as the intersection of finitely many surfaces. In addition, the randomized nature of forward search causes the edge to be discontinuous [Daniels et al. 2007]. Lipman's data-dependent MLS [2007] uses a different technique to segment the neighborhood, based on regularity detectors. After that, Lipman's projection uses piecewise polynomials which suffer from the same shortcomings of Fleishman's robust MLS. As we argued in Section 3.4, our proposed $\mathcal{P}_U$ is capable of exactly reconstructing any feature that is representable locally as a function in a function space $\mathcal{F}$. Gal et

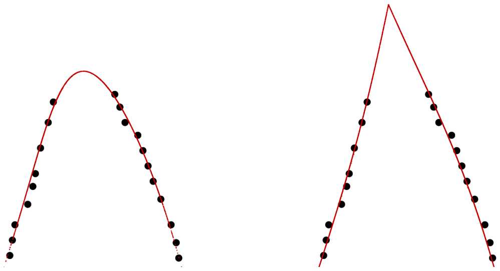


Figure 8: *A function space $\mathcal{F}$ that contains predetermined priors that fit the original surface can be used to complete missing details. On the left, $\mathcal{F}$ only contains smooth functions, so the gap is filled with a smooth reconstruction. On the right, $\mathcal{F}$ contains wedge shaped functions, so a sharp point is naturally reconstructed.*

al. [2007] first upsample a point set based on a database of local shape priors and then perform MLS reconstruction. Our technique does not require upsampling of the point set; we instead incorporate the priors into the function space and reconstruct the surfaces directly. Additionally, by allowing the function space to be closed under affine transformations (see Section 4.1), we allow priors to be transformed more freely, better fitting the local neighborhood.

In our work, we follow in the footsteps of Amenta and Kil [2004a] by showing that the failures near corners and edges are unrelated to sampling conditions – they are in fact intrinsic to the geometry of the surface from which the points are sampled. Our work also proposes an alternative way to define sharp features. Instead of using multiple steps, our technique is based on a conceptually simple modification of Levin's non-linear projection.

Our technique has some limitations. First, the resulting optimization of $\mathcal{P}_U$ is inherently more complex than that of $\mathcal{P}_L$. This means the optimizer will necessarily require more work to reach the optima. We have, however, observed that neighboring points tend to generate similar local reference frame and approximations. One way to mitigate this shortcoming might be to exploit this coherence to speed up successive projections. In addition, similarly to Guennebaud and Gross [2007], the more complicated energy field might lead to spurious minima (manifested in their work as extra zeros). As in their case, we have not found this to be a signficant issue in practice. It remains an important topic to investigate the conditions under which $\mathcal{P}_U$ is convex.

7 Conclusions and Future Work

In this paper, we have proposed a unified MLS projection operator. It is close in spirit to Levin's $\mathcal{P}_L$ and Lancaster and Salkauska's seminal work [2003; 1981], and it allows the use of rich function spaces that contain functions relevant to the surface being reconstructed. It is a natural extension to Levin's approach, and addresses a key shortcoming of the technique. We have provided an alternate presentation of Levin's proof that $\mathcal{P}_L$ is a projection, which naturally extends to our more general operator.

We believe that our approach has many applications. Our unified projection allows manifolds to be defined on embedding spaces richer than R^3. For example, it should be possible to reconstruct color information by operating on the product space of color and geometric information. The same technique should be applicable to texture coordinates, and other fields associated to the manifold. Another possibility is to investigate function spaces that are closed under a wider range of transformations — in particular, non-rigid transformations. This can potentially allow much richer spaces without having to create complicated projection definitions.

References

ADAMSON, A., AND ALEXA, M. 2003. Approximating and intersecting surfaces from points. In *Proc. Symposium on Geometry Processing*, 230–239.

ALEXA, M., AND ADAMSON, A. 2004. On normals and projection operators for surfaces defined by point sets. In *Proc. Symposium on Point-Based Graphics 2004*, 149–155.

ALEXA, M., AND ADAMSON, A. 2007. Interpolatory point set surfaces - convexity and hermite data. Tech. rep., T.U. Berlin.

ALEXA, M., BEHR, J., COHEN-OR, D., FLEISHMAN, S., LEVIN, D., AND SILVA, C. T. 2001. Point set surfaces. In *Proc. IEEE Visualization*, 21–28.

AMENTA, N., AND KIL, Y. J. 2004. The domain of a point set surface. In *Proc. Symposium on Point-Based Graphics*, 139–147.

AMENTA, N., AND KIL, Y. J. 2004. Defining point-set surfaces. *ACM Transactions on Graphics 23*, 3, 264–270.

AMENTA, N., BERN, M., AND KAMVYSSELIS, M. 1998. A new voronoi-based surface reconstruction algorithm. *Proc. ACM SIGGRAPH*, 415–422.

BREMER, P.-T., AND HART, J. C. 2005. A sampling theorem for MLS surfaces. In *Proc. Symposium on Point-Based Graphics*, 47–54.

CURLESS, B., AND LEVOY, M. 1996. A volumetric method for building complex models from range images. In *Proc. ACM SIGGRAPH*, 303–312.

DANIELS, J., HA, L., OCHOTTA, T., AND SILVA, C. 2007. Robust smooth feature extraction from point clouds. In *Shape Modeling International*, 123–133.

DEY, T. K., AND SUN, J. 2005. An adaptive MLS surface for reconstruction with guarantees. In *Proc. Symposium on Geometry Processing*, 43–52.

FLEISHMAN, S., COHEN-OR, D., ALEXA, M., AND SILVA, C. T. 2003. Progressive point set surfaces. *ACM Transactions on Graphics 22*, 4, 997–1011.

FLEISHMAN, S., COHEN-OR, D., AND SILVA, C. T. 2005. Robust moving least-squares fitting with sharp features. *ACM Transactions on Graphics 24*, 3, 544–552.

GAL, R., SHAMIR, A., HASSNER, T., PAULY, M., AND COHEN-OR, D. 2007. Surface reconstruction using local shape priors. In *Symposium on Geometry Processing 2007*.

GUENNEBAUD, G., AND GROSS, M. 2007. Algebraic point set surfaces. *ACM Transactions on Graphics 26*, 3 (July), 23:1–23:9.

HOPPE, H., DEROSE, T., DUCHAMP, T., MCDONALD, J., AND STUETZLE, W. 1992. Surface reconstruction from unorganized points. In *Proc. ACM SIGGRAPH*, 71–78.

KOLLURI, R. 2005. Provably good moving least squares. In *Proc. ACM-SIAM Symposium on Discrete Algorithms*, 1008–1017.

LANCASTER, P., AND SALKAUSKAS, K. 1981. Surfaces generated by moving least squares methods. *Mathematics of Computation 37*, 155, 141–158.

LEVIN, D. 2003. *Geometric Modeling for Scientific Visualization*. Springer-Verlag, ch. Mesh-independent surface interpolation, 37–49.

LIPMAN, Y., COHEN-OR, D., AND LEVIN, D. 2006. Error bounds and optimal neighborhoods for MLS approximation. In *Proc. Symposium on Geometry Processing*, 71–80.

LIPMAN, Y., COHEN-OR, D., AND LEVIN, D. 2007. Data-dependent MLS for faithful surface approximation. In *Symposium on Geometry Processing 2007*.

PAULY, M., KEISER, R., KOBBELT, L. P., AND GROSS, M. 2003. Shape modeling with point-sampled geometry. *ACM Transactions on Graphics 22*, 3, 641–650.

PFISTER, H., ZWICKER, M., VAN BAAR, J., AND GROSS, M. 2000. Surfels: Surface elements as rendering primitives. In *Proc. ACM SIGGRAPH*, 335–342.

PRESS, W., TEUKOLSKY, S., VETTERLING, W., AND FLANNERY, B. 1992. *Numerical Recipes in C*. Cambridge.

REUTER, P., JOYOT, P., TRUNZLER, J., BOUBEKEUR, T., AND SCHLICK, C. 2005. Surface reconstruction with enriched reproducing kernel particle approximation. In *Proceedings of Point-Based Graphics*.

RUSINKIEWICZ, S., AND LEVOY, M. 2000. Qsplat: A multiresolution point rendering system for large meshes. In *Proc. of ACM SIGGRAPH*, 343–352.

TURK, G., AND LEVOY, M. 1994. Zippered polygon meshes from range images. In *Proc. ACM SIGGRAPH*, 311–318.

WHEELER, M. D., SATO, Y., AND IKEUCHI, K. 1998. Consensus surfaces for modeling 3D objects from multiple range images. In *ICCV '98*, 917–924.

XIE, H., WANG, J., HUA, J., QIN, H., AND KAUFMAN, A. 2003. Piecewise g^1 continuous surface reconstruction from noisy point cloud via local implicit quadric regression. In *Proc. IEEE Visualization*, 259–266.