A Multi-Scale Tikhonov Regularization Scheme for Implicit Surface Modelling

Jianke Zhu, Steven C.H. Hoi and Michael R. Lyu
Department of Computer Science & Engineering
Chinese University of Hong Kong, Shatin, Hong Kong
{jkzhu,chhoi,lyu}@cse.cuhk.edu.hk

Abstract

Kernel machines have recently been considered as a promising solution for implicit surface modelling. A key challenge of machine learning solutions is how to fit implicit shape models from large-scale sets of point cloud samples efficiently. In this paper, we propose a fast solution for approximating implicit surfaces based on a multi-scale Tikhonov regularization scheme. The optimization of our scheme is formulated into a sparse linear equation system, which can be efficiently solved by factorization methods. Different from traditional approaches, our scheme does not employ auxiliary off-surface points, which not only saves the computational cost but also avoids the problem of injected noise. To further speedup our solution, we present a multi-scale surface fitting algorithm of coarse to fine modelling. We conduct comprehensive experiments to evaluate the performance of our solution on a number of datasets of different scales. The promising results show that our suggested scheme is considerably more efficient than the state-of-the-art approach.

1. Introduction

Machine learning has already achieved many successes in a broad range of application domains, such as pattern recognition, computer vision, and bioinformatics [14]. However, there is relatively little research attention on the area of 3D visual learning research. Recently, applications of machine learning techniques to 3D points cloud data processing are attracting increasing research interest, especially in the task of surface reconstruction. One of the successful paradigms applied in this area is to use kernel machines for implicit surface modelling, such as Support Vector Machines (SVM) [11, 19, 16].

The 3D objects are usually represented by triangulated meshes explicitly derived from 3D scattered data [17]. Most 3D data acquisition techniques, such as range scanners, stereo pairs and so on, always suffer from problems of incomplete data and noise. Recently, implicit surface models have received more and more attention for 3D object representation. The goal of implicit surface modelling is to estimate an embedding function f, whose zero level set $f^{-1}(0)$ implicitly defines the hyper-surface. Implicit surface models enjoy many advantages compared with traditional methods using explicitly triangulated meshes. The implicit models generate smooth surfaces and are able to repair holes and filter outliers in the data; these are often difficult to achieve with the explicit models. Moreover, a number of derivatives of the embedding function f will usually exist, which are useful for further analysis. The implicit surface model can even be used to reconstruct the time-varying scene [5].

Numerous approaches have been suggested for implicit surface modelling. One popular solution is to use local nature for inferring the implicit functions, such as level set models [20, 15], local surface models [7], geometric flow [9] or implicit surfaces interpolated from polygon data [13]. Due to the use of local nature for analysis, most of these methods often require normal information about the target surface in order to generate the implicit surfaces correctly. Recently, kernel machines have been proposed as another solution for implicit surface modelling [11]; the implicit models are typically represented by a mixture of radial basis functions, either fully supported [2] or compactly supported [8]. These methods usually also need the surface normal information, except in the case of recent works [11, 19].

Although machine learning techniques have been shown as a promising solution for implicit surface modelling, one important challenge is the high computational cost, which has so far received little consideration in previous work. The number of scattered points for building a 3D object is typically of the order of millions. Modelling such a 3D object by machine learning is usually related to a large-scale regression problem which often requires a very high computational cost. Storing and rendering implicit surfaces may also require high computational cost for a complex implicit model. Hence, a concise representation of the implicit model is vitally important for fast applications.

One way to reduce the computation cost is to avoid the use of normal information. The normal information is of-

ten engaged in traditional methods which usually generate two off-surface points along each normal vector of the surface. The off-surface points are used to avoid intersection with other parts and to make the fitting process simpler. For example, in the $\varepsilon-$SVR method [16], off-surface points are introduced to ensure non-trivial solutions of the optimization. However, the use of off-surface points results in an increase of the problem size, which will raise the computational cost. Moreover, using the normal information may introduce additional noise due to the inaccurate estimation of normal vectors by local methods. Hence, it is better to avoid the use of normal information. To reduce computational cost, another key consideration is to formulate an elegant optimization which can be solved very efficiently.

To this end, we present an efficient framework for implicit surface modelling based on a hierarchical Tikhonov regularization scheme, in which the optimization problem can be solved efficiently by factorization methods. Our solution needs no additional normal information, which not only keeps the problem size small, but also avoids introducing additional noise. Moreover, our multi-scale algorithm of coarse to fine fitting reduces the number of base points to attain a concise representation of the implicit model.

The rest of this paper is organized as follows. Section 2 reviews recent advances of implicit surface modelling using machine learning methods. Section 3 presents our proposed scheme for implicit surface modelling based on a regularization framework. Based on the regularization framework, an efficient solution is proposed by the Tikhonov regularization that can be solved efficiently by factorization methods. To deal with large-scale problems, a multi-scale fitting algorithm is suggested for coarse to fine fitting. Section 4 discusses the details of our experimental implementations and demonstrates our experimental results. Section 5 sets out our conclusion.

2. Related Work

There are only a few reports of work on implicit surface modelling using machine learning techniques. Most of these can be divided into two categories:

- **Not using normal.** Methods such as Slab SVM [11] and its extension [19] have recently shown promise for implicit surface modelling.

- **Using normal.** The representative method is the ε-SVR [16], in which the normal information is used to simplify the problem and avoid trivial solutions.

Slab SVM method is a modified one-class SVM derived by replacing the hinge loss function with the ε-insensitive loss function. More specifically, it can be formulated as an optimization problem as follows:

$$\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} (f(\mathbf{x}_i))_{\varepsilon} + \lambda \|f\|_{\mathcal{H}}^2 - \rho$$

where $\|f\|_{\mathcal{H}}^2$ acts as a regularizer in which $\mathcal{H}$ denotes a reproducing kernel Hilbert space (RKHS), $\lambda > 0$ is a regularization parameter, and ρ is a constant to avoid trivial solutions of the optimization. Typically it uses a fully supported radial basis functions (RBF) kernel, which usually engages a very heavy computational cost. In one reported study [11], it required about 2 hours to build and render an implicit model from around 40K scattered points. Furthermore, one disadvantage of Slab SVM is that it is not able to fix holes conveniently since it adopts fixed kernel widths.

The work in [19] proposed a treatment for the Slab SVM approach by introducing extra regularization terms and multi-scale basis functions. Moreover, the ε-insensitive loss function was replaced by a 2-norm loss function. More specifically, the optimization problem becomes:

$$\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} f(\mathbf{x}_i)^2 + \lambda \|f\|_{\mathcal{H}}^2 - G(f)$$

where $G(f)$ denotes the summation of energy and gradients terms. Although the optimization problem can be solved efficiently by the way of an equivalent eigenvalue problem, this approach still took about 40 minutes to solve a problem with around 35K samples, according to the study in [19].

Both Slab SVM and its extension are so far not computationally efficient for industrial applications. Recently, [16] proposed another more efficient method using a modified support vector regression. They suggested a Gauss-Seidel method to solve the quadratic program with a positive definite kernel matrix and a box constraint. However, the Gauss-Seidel approach usually has a slow convergence rate and may not always guarantee the correct solution. Moreover, they use the normal information, which will always increase the problem size.

In contrast to previous work, our proposed scheme is based on a hierarchical Tikhonov regularization scheme, in which the optimization problem can be solved very efficiently by factorization methods. In addition to computationally highly competitive performance, our proposed multi-scale solution, which does not use additional normal information, is free from injected noise and enjoys a concise representation of the implicit surface models.

3. Implicit Surface Modelling

3.1. Theoretical Foundation

In general, building an implicit surface model can be regarded as a regression problem, which approximates a multivariate function from scattered data. Such a problem is

usually ill-posed. One efficient way to solve the problem is based on the theoretical framework of regularization networks [4]. This typically formulates the issue as a variation of finding the embedded function f to solve the following minimization problem:

$$\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} V(y_i, f(\mathbf{x}_i)) + \lambda \|f\|_{\mathcal{H}}^2 \qquad (1)$$

where $V(\cdot, \cdot)$ is the loss function, and $(\mathbf{x}_i, y_i)_{i=1}^n$ are the n pairs of samples. There are various choices for the loss function. For example, with the L_2 norm, it becomes the classical L_2 regularization networks, which is also known as the Tikhonov regularization:

$$V(y, f(\mathbf{x})) = (y - f(\mathbf{x}))^2 \qquad (2)$$

If $V(\cdot, \cdot)$ is an ε-insensitive function, the problem turns out to be the ε-SVR

$$V(y, f(\mathbf{x})) = (y - f(\mathbf{x}))_\varepsilon \qquad (3)$$

For the implicit surface modelling task, our goal is to find an embedding function f to approximate the signed distance transformation function. According to the definition of the signed distances, points on the surface lie in the zero level-set $f^{-1}(0)$. Namely the value of y_i is zero for all samples, hence we reformulate Eqn. (2-3) into the following optimizations:

$$\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} f(\mathbf{x}_i)^2 + \lambda \|f\|_{\mathcal{H}}^2 \qquad (4)$$

$$\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} (f(\mathbf{x}_i))_\varepsilon + \lambda \|f\|_{\mathcal{H}}^2 \qquad (5)$$

Given the above regularization networks, much previous work can be generalized into this framework. For example, the Slab SVM [11] is equivalent to Eqn. (5) with a bias term ρ, and the optimization problem of its extension [19] can be viewed as Eqn. (4) with an extra regularization term $G(f)$. Our proposed solution is based on the same theoretical framework. However, we use the Tikhonov regularization scheme given in Eqn. (4) without any additional term, in which the optimization problem can be solved very efficiently by factorization methods.

Remark. The additional regularization terms in the previous approaches are usually to avoid the triviality issue, which is not a problem in our scheme. Another function of $G(f)$ in [19] is to control the tradeoff between smoothness and the fidelity to the data. In our scheme, we solve this problem by a hierarchical fitting solution, which is much more flexible and effective than the previous complicated approach.

3.2. A Tikhonov Regularization Approach

Since all scattered data lie on the zero level-set, the Tikhonov regularization scheme for the implicit surface modelling turns out to be a one-class problem. According to the Represented theorem [12], any $f \in \mathcal{H}$ minimizing the regularized risk function in Eqn. (1) will have a representation of the form

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i k(\mathbf{x}, \mathbf{x}_i) + b \qquad (6)$$

where b is an offset term, and $k(\cdot, \cdot)$ is the kernel function. Substituting the representation of Eqn. (6) into Eqn. (4), we get the convex differentiable object function $E(\alpha)$ of the variable $\alpha = [\alpha_1, \cdots, \alpha_n]^T$ as follows:

$$E = \min_{\boldsymbol{\alpha} \in R^n, \lambda} \frac{1}{n} (K\boldsymbol{\alpha} + \mathbf{e}^T b)^T (K\boldsymbol{\alpha} + \mathbf{e}^T b) + \lambda \boldsymbol{\alpha}^T K \boldsymbol{\alpha} \quad (7)$$

where $\mathbf{e}$ denotes a vector with all elements equal to one. The derivatives of $E(\boldsymbol{\alpha})$ with respect to the variable α vanish for optimality:

$$\frac{\partial E}{\partial \boldsymbol{\alpha}} = \frac{1}{n} (K\boldsymbol{\alpha} + \mathbf{e}^T b)^T K + \lambda K \boldsymbol{\alpha}$$

which leads to the following solution:

$$\boldsymbol{\alpha} = -b(K + n\lambda I)^{-1} \qquad (8)$$

where the kernel matrix $K \in R^{n \times n}$ is symmetric positive definite. The kernel function is either fully supported or compactly supported. The support vectors of compactly supported functions usually cover only a minor part of the data. Thus, they can deal with very large-scale problem efficiently, making full use of sparsity of the system as local portions of much larger problems. Using compactly supported radial basis functions will pay off with respect to computational efficiency, while maintaining a comparable level of accuracy in the reproduction. Typically, the requirements of computational cost and storage can be reduced by using sparse basis functions of various widths. To enable the sparsity, we use the following Wu function [10] as a kernel in our scheme:

$$k(\mathbf{x}_1, \mathbf{x}_2) = k(r(\mathbf{x}_1, \mathbf{x}_2)) = (1-r)_+^4 (4+16r+12r^2+3r^3)$$

where $r(\mathbf{x}_1, \mathbf{x}_2) = \frac{\|\mathbf{x}_1 - \mathbf{x}_2\|}{\sigma}$, in which $\sigma > 0$ is the size of compact support.

The problem in Eqn. (8) is a linear equation, which can be solved by singular value decomposition (SVD), LU factorization, Cholesky factorization, or LDL^T factorization [1]. The SVD method is usually quite stable. But it is not suitable for this problem which is usually a large system with a high sparsity ratio. Table 1 shows a comparison of

three factorization algorithms. We can see that the Cholesky and the LDL^T methods are more efficient than the LU factorization when the matrix is symmetric and positive definite. Since the kernel matrix K is sparse, the computational cost is determined by the number of base points n and the number of non-zero elements of K. Normally a fast nearest neighbor searching method can be used to compute the kernel expansion in Eqn. (6), in which the total number of nearest neighbors for all base points is equal to the number of non-zero elements of K. Such an approach will usually decrease the complexity of computing K from $O(n^2)$ to $O(n \log n)$. In this paper, we adopt the Cholesky and the LDL^T factorization algorithms to solve the optimization problem.

Table 1. Comparison of factorization algorithms.

ALGORITHM	REQUIREMENT	COMPLEXITY
LU	NON-SINGULAR	$\sim (2/3)n^3$ FLOPS
CHOLESKY	SPD	$\sim (1/3)n^3$ FLOPS
LDL^T	SPD	$\sim (1/3)n^3$ FLOPS

SPD denotes "Symmetric and Positive Definite."

3.3. A Multi-Scale Implicit Surface Modelling Algorithm

The method using compactly supported kernels always suffers from the problem that the approximated function is valid only in a bounded region. Hence, a multi-scale approach is a necessary treatment to achieve a global approximation.

In order to decrease the computational cost and storage requirements, the total number of base points should be reduced. Intuitively, the minimum number of base points can be attained when the compact support size σ_i of each base point $\mathbf{x}_i$ takes different values:

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i k_{\sigma_i}(\mathbf{x}, \mathbf{x}_i) + b$$

However, it is almost impossible to tune such a large number of variables in the above formulation.

Instead of putting all samples into a large training set with a fixed compact support size σ, we divide them into m subsets: $S = S_1 \cup S_2 \cup \cdots \cup S_m$. Each subset S_i with n_i samples is trained hierarchically. A recent study [18] has proved that if the regularization parameter is fixed, the compact support size decreases when the number of samples increases. Thus, we can construct the subsets through a subdivision scheme, in which the number of samples assigned in the subset increases with the scale. The compact support size σ_i for each subset S_i will decrease, and its value can be estimated from the previous level by a fixed damping ratio η ($0 < \eta < 1$). Therefore, the hyper-surface $f(x)$ can be represented as follows:

$$f(\mathbf{x}) = \sum_{i=1}^{n_1} \alpha_{1i} k_{\sigma_1}(\mathbf{x}, \mathbf{x}_i) + \sum_{i=1}^{n_2} \alpha_{2i} k_{\sigma_2}(\mathbf{x}, \mathbf{x}_i)$$
$$+ \cdots + \sum_{i=1}^{n_m} \alpha_{mi} k_{\sigma_m}(\mathbf{x}, \mathbf{x}_i) + b \qquad (9)$$

where $n_0 < n_1 < \cdots < n_m$, $\sigma_i = \eta \sigma_{i-1}$ ($2 \leq i \leq m$). Let h_l denote the embedded function at the level l:

$$h_l(\mathbf{x}) = \sum_{i=1}^{n_l} \alpha_{li} k_{\sigma_l}(\mathbf{x}, \mathbf{x}_i)$$

Substituting the above equations into Eqn. (6), we obtain a concise form:

$$f(\mathbf{x}) = \sum_{j=1}^{m-1} h_j(\mathbf{x}) + b + y_m(\mathbf{x})$$

where the value of $y_m(\mathbf{x})$ is the fitting residual. For each level $l = 2, 3, \ldots, m$, we have

$$\sum_{j=1}^{l-1} h_j(\mathbf{x}) + b + y_l(\mathbf{x}) = 0$$

$$y_l(\mathbf{x}) = -\sum_{j=1}^{l-1} h_j(\mathbf{x}) - b$$

Hence, the optimization problem at the level l can be formulated as:

$$\min_{h_l \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} (y_l(\mathbf{x}_i) - h_l(\mathbf{x}_i))^2 + \lambda \|h_l\|_{\mathcal{H}}^2 \qquad (10)$$

This is equivalent to minimizing the following objective of E_l:

$$E_l = \min_{\boldsymbol{\alpha}_l, \lambda_l} \frac{1}{n_l} (\mathbf{y}_l - K_l \boldsymbol{\alpha}_l)^T (\mathbf{y}_l - K_l \boldsymbol{\alpha}_l) + \lambda_l \boldsymbol{\alpha}_l^T K_l \boldsymbol{\alpha}_l$$

which leads to the following solution:

$$\boldsymbol{\alpha}_l = (K_l + n_l \lambda_l I)^{-1} \mathbf{y}_l \qquad (11)$$

Compared with a traditional SVM approach, the above Tikhonov regularization scheme may not have a good performance of basis shrinkage, i.e., only a few portions of α_{l_i} will vanish in the solution. To tackle this problem, we suggest a treatment in the multi-scale solution. At each level l, each data point $\mathbf{x}$ is tested for removal from the working set if the residual $y_l(\mathbf{x})$ reaches the setting precision. For the level l, we assign the precision with the value of $p \cdot \sigma_l$ and denote the product of $n_l \lambda_l$ as γ_l.

Table 2. Results of computational cost on various datasets: number of data points, number of scales m, total number of base points and average fitting time in seconds for ε-SVR and proposed methods, and relative fitting accuracy.

DATASET	#POINTS	#SCALE m	#BASES-1	ε-SVR[S]	#BASES-2	CHOLESKY[S]	LDL$^{\mathrm{T}}$[S]	ACCURACY
HAND	39.2K	4	37.0K	28.1	17.4K	1.7	1.7	5×10^{-4}
AMADILLO	173.0K	6	234.4K	131.2	121.9K	25.1	24.4	2×10^{-4}
BUNNY	28.0K	5	25.0K	17.3	19.0K	1.1	1.1	8×10^{-4}
SQUIRREL	76.3K	6	133.1K	120.7	70.1K	17.2	17.0	6×10^{-4}
IGEA	72.5K	6	63.9K	22.3	42.1K	2.9	2.8	5×10^{-4}
KNOT	28.7K	4	38.0K	37.7	12.3K	1.0	0.9	9×10^{-4}
DINO	56.2K	5	71.1K	33.4	42.9K	2.8	2.8	5×10^{-4}
FELINE	199.5K	6	202.8K	114.3	99.9K	11.6	11.5	4×10^{-4}
DRAGON	437.6K	7	346.3K	365.9	201.9K	79.8	77.9	8×10^{-4}

#BASES-1 denotes the number of base points by ε-SVR; #BASES-2 denotes the number of base points by the two factorization methods.

Remark. One can easily modify the algorithm above to introduce off-surface points by adding them into the training set with estimated signed distance values. More specifically, one can modify the second step of the algorithm as:

$$\boldsymbol{\alpha}_1 = (K_1 + \gamma_1 I)^{-1}(\mathbf{y} - \mathbf{e}^T b)$$

and keep other parts of the original algorithm unchanged. Hence, our scheme can deal with problems either with or without normal information conveniently.

Empirically, in the above algorithm, the initial kernel parameter σ is typically set to the radius of the modelling object, and b is usually set to σ. The damping ratio η is equal to 0.5. One can see that the total number of base points is controlled by the number of scales m and the precision threshold p. The function smoothness is related to the weight γ_l for the regularization term.

4. Experiments

In this section, we discuss the details of our experimental implementation and report the results of performance evaluation of implicit surface modelling. We use a number of different 3D datasets in the experiments, in which most of them are from the Stanford 3D Scanning Repository. All the experiments reported in this paper were carried out on a Pentium-4 3.0 GHz PC with 2GB RAM. A standard marching cube algorithm is employed to visualize the implicit surfaces [6], which roughly demonstrates the embedding function over the entire view. The CHOLMOD package [3] is used for both the Cholesky and the LDL$^{\mathrm{T}}$ factorizations in our experiment for solving the optimization problems. We also implemented a multi-scale gradient based ε-SVR method used in [11], which is regarded as the state-of-the-art approach.

4.1. Evaluation of Computational Performance

To examine the performance of computational efficiency of our proposed scheme, we evaluate the computational cost of our methods on a number of different datasets. Table 2 shows the results of time cost of different methods on nine

3D datasets with different sizes of data points. In the table, the relative fitting accuracy is calculated from the number of scales m and the given precision p. The time cost of each method on every dataset includes the time for generating the multi-scale subsets and for modelling the implicit surfaces. From the experimental results, we can see that our proposed factorization methods significantly outperform the ε-SVR method by gradient based optimizations in all test cases. The two factorization methods perform very closely; the LDL$^{\mathrm{T}}$ method is marginally better than the Cholesky method.

More specifically, looking into the results in detail, we observe that our solution can build the implicit surface model of the Stanford bunny (28K points) with 19K base points within 2 seconds. This is computationally highly competitive relative to previous work. For example, the state-of-the-art commercial solution of Fast RBF needed around 70 seconds for modelling the bunny data with 29.7K base points [2]. Compared with the performance reports in [16], our solution is significantly more efficient than the ε-SVR method.

4.2. Evaluation of Surface Fitting Performance

In addition to its highly competitive computational performance, we want to examine whether our multi-scale fitting method by the hierarchical Tikhonov regularization scheme can obtain competitive implicit surface models when applied to different data sets.

Multi-Scale Fitting. One important advantage of our solution is the multi-scale fitting. Figure 1 illustrates a multi-scale fitting example based on our hierarchical Tikhonov regularization scheme. From the results, we can see that the surface is smoother when the compact support size σ is larger. The base points of the high level represents the detail information of the surface. This multi-scale fitting solution can be applied to many applications of multi-resolution surface reconstruction.

Regularization. The regularization factor plays an important role in the implicit surface modelling. When the regularization term increases, the fitting function will become

Figure 1. Illustration of a multi-scale fitting example by the Tikhonov regularization approach. Armadillo (170K points, 24.4 seconds) from the Stanford 3D Scanning Repository is employed as the point cloud data. The rendered implicit surfaces are plotted at six levels.

smoother, while the fitting error will usually increase. Generally speaking, using an appropriate regularization term can avoid the over-fitting issue of implicit surface modelling.
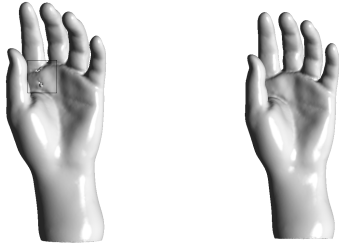

Figure 2. Example of the over-fitting problem. The left side is the over-fitting case without using regularization.
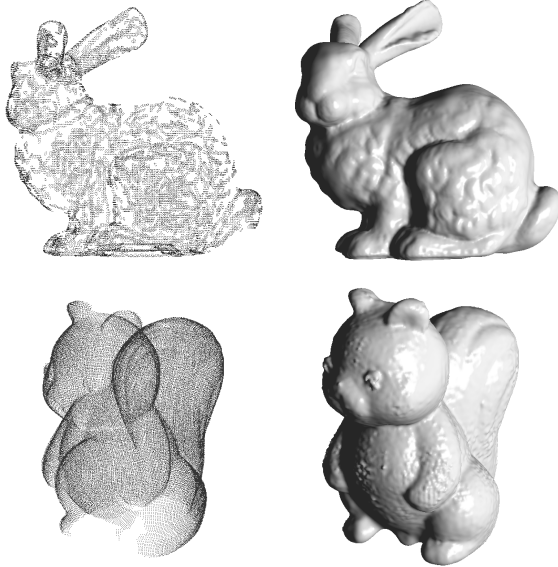

Figure 3. Modelling the cases of incomplete data. The bunny (28K points, 1.1 seconds) and the squirrel (76K points, 17 seconds) are studied.

Figure 2 shows an example of avoiding the over-fitting problem. The left sub-figure shows the overfitting case without regularization in which holes occur in the intersection of fingers. To tackle this problem, one can increase the regularization term at the low level in the multi-scale fitting process. The fitting error caused by the large regularization term at the low levels could be compensated at higher lev-

els. The right sub-figure shows the correct results using a proper regularization parameter.

Interpolation of Incomplete Data. One important advantage of implicit surface models is able to interpolate incomplete data conveniently so that it can repair the holes of surfaces due to missing data. Figure 3 shows the bunny and squirrel examples in which parts of the 3D data are missing in the training sets shown in the left sub-figure. By applying our proposed technique, we can reconstruct the smooth surface without visual artifacts shown in the right sub-figure. Moreover, our scheme is able to deal with irregularly sampled cases. Figure 4 shows an example of an irregularly sampled 3D object. We can see that our technique can render the correct 3D surface without visual artifacts. Finally, Figure 5 shows several large-scale examples modelled by our technique.
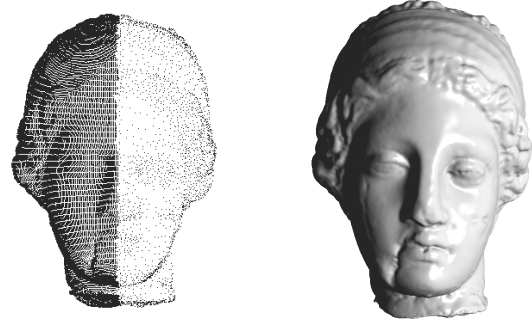

Figure 4. Modelling an irregularly sampled Stanford Igea (73K points, 2.8 seconds). The right part of the original Igea is 90% decimated.

5. Conclusion

In this paper we presented a novel and efficient solution for the implicit surface modelling using machine learning techniques. We first outlined a theoretical framework of regularization networks in which the findings of several previous studies can be considered as special cases within the given regularization framework. Based on the solid theoretical framework, we suggested to tackle the implicit surface modelling problem using a Tikhonov regularization scheme. The optimization problem of the Tikhonov regularization scheme can be formulated into a sparse linear equation system, which can be efficiently solved by factorization methods. To further save computational cost and achieve good sparsity, we proposed a multi-scale fitting algorithm

Figure 5. Examples of large-scale implicit surface modelling. The Dino (56K points, 2.8s) with complex edge structures is modelled by 43K bases, and the Caltech feline (200K points, 11.5s) is modelled by 100K bases. The Stanford dragon contains more than 400K data points.

for the implicit surface modelling problem, which can reduce the total number of base points of the resulting models. Our empirical evaluations on a number of datasets of different scales demonstrated that our proposed method is more efficient than the state-of-the-art approaches. In addition to the advantage of computational efficiency, our solution also solves several challenging issues in surface modelling and reconstruction, such as multi-resolution surface reconstruction, the absence of normal information, incomplete data interpolation, and irregularly sampled cases.

Acknowledgments

References

[1] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. 3

[2] J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans. Reconstruction and representation of 3d objects with radial basis functions. In *SIGGRAPH '01*, pages 67–76, 2001. 1, 5

[3] T. A. Davis. Algorithm 849: A concise sparse Cholesky factorization package. *ACM Transactions on Mathematical Software*, 31(4), Dec. 2005. 5

[4] T. Evgeniou, M. Pontil, and T. Poggio. Regularization networks and support vector machines. *Advances in Computational Mathematics*, 13:1–50, 2000. 3

[5] B. Goldlücke and M. A. Magnor. Space-time isosurface evolution for temporally coherent 3d reconstruction. In *CVPR (1)*, pages 350–355, 2004. 1

[6] T. Lewiner, H. Lopes, A. W. Vieira, and G. Tavares. Efficient implementation of Marching Cubes cases with topological guarantees. *Journal of Graphics Tools*, 8(2):1–15, 2003. 5

[7] Y. Ohtake, A. Belyaev, M. Alexa, G. Turk, and H. Seidel. Multi-level partition of unity implicits. *ACM Trans. Graph.*, 22(3):463–470, 2003. 1

[8] Y. Ohtake, A. Belyaev, and H.-P. Seidel. A multi-scale approach to 3d scattered data interpolation with compactly supported basis functions. In *SMI'03*, page 292, 2003. 1

[9] P. Savadjiev, F. P. Ferrie, and K. Siddiqi. Surface recovery from 3d point data using a combined parametric and geometric flow approach. In *EMMCVPR*, pages 325–340, 2003. 1

[10] R. Schaback. Creating surfaces from scattered data using radial basis functions. *Mathematical methods for curves and surfaces*, pages 477–496, 1995. 3

[11] B. Schölkopf, J. Giesen, and S. Spalinger. Kernel methods for implicit surface modeling. In Y. W. L. B. Saul, L.K., editor, *Advances in Neural Information Processing Systems*, volume 17, pages 1193–1200. MIT Press, 2005. 1, 2, 3, 5

[12] B. Schölkopf, R. Herbrich, and A. Smola. A generalized representer theorem. In *Computational Learning Theory, NeuroCOLT*, pages 416–426, 2001. 3

[13] C. Shen, J. F. O'Brien, and J. R. Shewchuk. Interpolating and approximating implicit surfaces from polygon soup. In *ACM SIGGRAPH 2004*, pages 896 – 904, 2004. 1

[14] A. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, 14(3):199–222, 2004. 1

[15] J. E. Solem and A. Heyden. Reconstructing open surfaces from unorganized data points. In *CVPR (2)*, pages 653–660, 2004. 1

[16] F. Steinke, B. Schölkopf, and V. Blanz. Support vector machines for 3d shape processing. In *EUROGRAPHICS 05*, volume 24, pages 285 – 294, 2005. 1, 2, 5

[17] D. Tubic, P. Hébert, and D. Laurendeau. 3d surface modeling from range curves. In *CVPR (1)*, pages 842–849, 2003. 1

[18] R. Vert and J.-P. Vert. Consistency and convergence rates of one-class svm and related algorithms. In *Advances in Neural Information Processing Systems*. MIT Press, 2006. 4

[19] C. Walder, O. Chapelle, and B. Schölkopf. Implicit surface modelling as an eigenvalue problem. In S. W. De Raedt, L., editor, *ICML 2005*, pages 937 – 944, 2005. 1, 2, 3

[20] H.-K. Zhao, S. Osher, and R. Fedkiw. Fast surface reconstruction using the level set method. In *VLSM'01*, page 194, 2001. 1