Super-resolution of Point Set Surfaces using Local Similarities

Azzouz Hamdi-Cherif[1,2] Julie Digne[1,3] Raphaëlle Chaine[1,3]

[1] Univ Lyon - LIRIS - CNRS UMR5205, [2] INSA de Lyon, [3] Université Lyon 1

Abstract

3D scanners provide a virtual representation of object surfaces at some given precision that depends on many factors such as the object material, the quality of the laser-ray or the resolution of the camera. This precision may even vary over the surface, depending for example on the distance to the scanner which results in uneven and unstructured point sets, with an uncertainty on the coordinates. To enhance the quality of the scanner output, one usually resorts to local surface interpolation between measured points. However, object surfaces often exhibit interesting statistical features such as repetitive geometric textures. Building on this property, we propose a new approach for surface super-resolution that detects repetitive patterns or self-similarities and exploits them to improve the scan resolution by aggregating scattered measures. In contrast with other surface super-resolution methods, our algorithm has two important advantages. First, when handling multiple scans, it does not rely on surface registration. Second, it is able to produce super-resolution from even a single scan. These features are made possible by a new local shape description able to capture differential properties of order above 2. By comparing those descriptors, similarities are detected and used to generate a high-resolution surface. Our results show a clear resolution gain over state-of-the-art interpolation methods.

1. Introduction

Object acquisition devices have developed recently following two distinct trends. On the one hand, mainstream devices enable a fast shape acquisition but at a limited resolution and, on the other hand expensive professional 3D scanners are able to acquire high quality point clouds but at a slower rate. This dual development raises new challenges such as the production of high resolution data from low-resolution acquisitions. This process called super-resolution (SR) has been extensively studied in the context of signal and image processing but rarely tackled in geometry processing. The super-resolution problem should be distinguished from the inpainting problem. Both aim at adding geometric informations at locations where no such information is available. However the scale for super-resolution is much smaller than the one of an inpainting application: instead of filling large identified surface holes, the point cloud is upsampled by inferring plausible geometric information.

Multiple scans super-resolution aggregate several measures of the same object taken from very close viewpoints. It requires aligning the measures using global registration and merging the information yielding a mathematically well-posed problem. On the contrary, single scan super-resolution aims at producing a high resolution point cloud with insufficient data. Although this problem is ill-posed, solutions can be found by making some assumptions on the nature of the surface, or by regularizing the solution (e.g. using Total Variation or Sparse Decomposition).

In this paper, we propose to solve this problem under the self-similarity hypothesis: we assume that the surface contains geomet-

ric textures that are repeated over the surface. We also assume that similar details at a given small scale reflect a similarity at a higher scale. In a nutshell our method detects and exploits local shape similarities in a single scan to synthesize high-resolution point sets. The intuition is the following: if a surface contains repeated details, one can consider that such a detail is acquired several times from different viewpoints. By analyzing those detail acquisitions jointly, one can enhance them (Figure 1). Figure 2 illustrates the principle of our method on a 1D example. Importantly enough, the resulting enhancement is close to the actual surface if the ground assumptions is fullfiled, otherwise it will yield a plausible self-similar surface. The self-similar hypothesis can be justified for both natural surfaces, because of the texture of the material (e.g. bark of a tree), or man-made object surfaces, because of the manufacturing tools. Glasner et al. [GBI09] showed that over 90% patches from natural images (out of 300 images) have more than 9 other similar patches within the same image. While such a thorough analysis has not been performed for surfaces yet, image self-similarity is strongly correlated with object surface self-similarity.

Using a very close definition of patches and similarity between patches, in our experiments each patch has 70 similar patches on 60000 patches in a shape in average. Analyzing surface details requires to extract relevant information in a robust way to be able to compare these variations. While usual descriptors express surface variations over a local tangent plane, giving hence a high importance to curvatures, we rather focus on finer details by expressing what lies above an order 2 surface. Thus details will be found sim-

Figure 1: *Super-resolution of a single scan of the Maya point set. Left: initial scan, right: super-resolution. For visualization purposes, both are reconstructed with Poisson Reconstruction.*

ilar even if they lie in areas with different curvatures. In this paper, we focus on the case of surfaces described by point sets endowed with normal information. Point sets are indeed the raw output of most 3D scanners, either as a sparse depth map or a complete point cloud.

To summarize, our contributions are the following:

- A local descriptor able to capture differential properties with a higher degree than the curvature.
- A non-local way to combine descriptors to generate high resolution information based on their similarities
- The possibility to perform single scan super-resolution as well as multiple scan super-resolution without any prior registration in the same framework.

2. Related Work

2.1. Super-resolution for 2D images

Image super-resolution methods can be classified into 3 categories: *classical super-resolution methods*, *example-based super-resolution*, and *learning-based super-resolution*.

Classical super-resolution assumes that a given image is the result of a process taking as input a high resolved version of this image. This process is mathematically modeled as a succession of linear transformations starting from the high resolved image and leading to a low resolved image. The estimated inverse process is then able to produce a super-resolution image from many low resolved images corresponding to the same scene with a slightly different viewpoint [EF97, EHO01, BK02]. Despite an elegant formalization, these methods offer only a small resolution gain factor, rarely exceeding 2.

Example-based super-resolution uses a database of patch pairs associating a low resolution image patch p_l to its corresponding high resolution image patch p_h. The idea is to replace each patch p by a more resolved patch p_{h0} such that its corresponding low-resolved version p_{l0} is the patch closest to p in the database [PH93]. To overcome the artefacts caused by a patch replacement, Freedman et al. Freeman et al. [FJP02] add a spatial consistency term thanks to a Markov Random Field. Ebrahimi et al. [EV07, FF11] notice that under a small resolution decrease, image singularities remain similar to themselves, so that upscaling can be approximated by patch growth. A drawback of this elegant method is that

the scale and curvature of very small details may be changed by this process. Finally, Glasner et al. [GBI09] use a single image and simulate a set of acquisitions by exploiting the inherent similarities of the scene at different scales to form a patch database used for example-based super-resolution. Our method will also be able to use a single measure (a single scan). However it will not rely on a multiscale decomposition, since the acquisition process does not involve a perspective projection, and will work at a single resolution.

Learning-based super-resolution statistically models the resolution loss process. Peleg et al. [PE14] compute two distinct sparse representations from a training set containing corresponding low and high resolution image pairs and learn the relationship between those representations. Thus, from the sparse representation of an image patch, the method is able to predict the sparse representation of the corresponding high resolved image patch. Similarly, Yang et al. [YWHM10] learn the low and high-resolution dictionaries jointly, such that two corresponding patches have the same representation over their respective dictionaries. The statistical evolution of the gradient profile with a change of resolution has been also studied to provide a super-resolution prior [SSXS08].

2.2. Super-resolution and interpolation for 3D surfaces

Although super-resolution for images and signals has been extensively studied, it is far less explored for 3D surfaces. A noticeable exception is the work of Kil et al. [KMA06] where classical super-resolution is applied to surfaces. This algorithm aligns input depth images acquired from slightly different viewpoints and computes a refined height map by averaging registered points on a finer grid. After such a process, the systematic errors of the scanning devices can be removed using a probabilistic model [AKSA09]. This method being based on merging information from additional scans, it cannot be adapted to the single scan case. With a single scan, no additional information is available, and the process reduces to a simple linear interpolation with Gaussian weights, which does not produce additional details. Super-resolution is strongly linked to surface interpolation or surface resampling. For example, it is possible to use the Moving Least Squares framework to interpolate points from the input low-resolution scan. Among others, [OGG09] and [GG07] are two state-of-the-art point set surfaces definitions that are feature and detail preserving. Our method differs radically from traditional interpolation methods. Indeed, the latter use purely geometric information such as differential properties (estimated at a given scale) to perform local interpolations. On the contrary, in the super-resolution case, the added information is driven by the joint analysis of surface areas sharing similar details. In this paper, we defend the idea that this similarity can be measured as the repeatability of a given detail all over the surface. We will however compare our results with these surface interpolation methods, bearing in mind that interpolation adds points based only on local neighborhoods while our method aggregates informations from all over the surface.
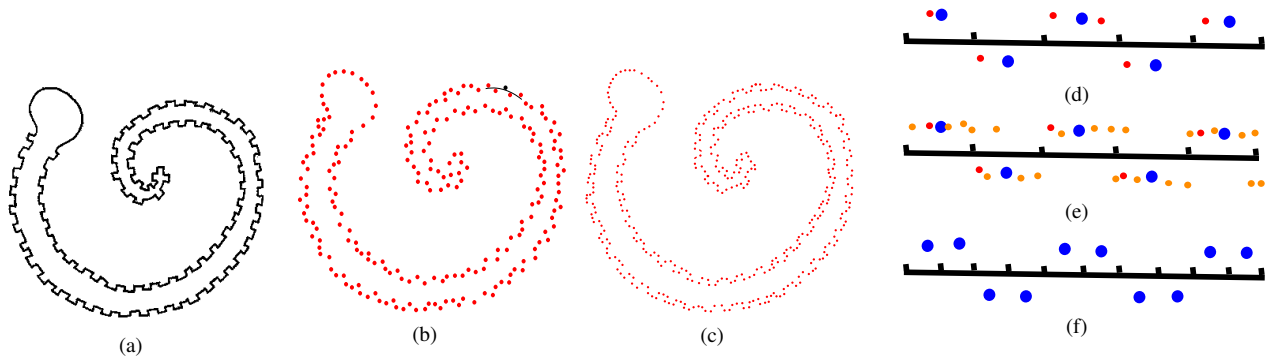
Figure 2: *Super-resolution of an input shape with highly repetitive geometric texture. a) Underlying shape to be sampled by an acquisition device; b) Low resolved input sampling of the shape and local approximation with a quadric at each point; geometric texture is the residue over the quadric; c) Super-resolved resampling using our method (fusion of super-resolved local patches). Right column: Generation of the super-resolved patches. d) Construction of a local descriptor of the residue over a low resolution grid corresponding to the unfolded quadric; blue points represent the height values estimated at bin centers, red points are the input points; e) Similar descriptor points are added (orange points) to the input points (in red) of the local descriptor; f) A super-resolved descriptor is computed from the set of red and orange points.*

2.3. Extracting surface similarities

In order to analyze surface similarities, one must encode local neighborhood variations in a robust and informative way. Although, local surface descriptors are extensively used for shape registration [JH99, SA01], they are far less exploited for shape analysis. As an exception, in the context of shape interactive editing, [GTB14] uses descriptors to deform simultaneously similar parts of the shape. However, it relies on spin images [JH99] which cannot be used to resample surface points from the descriptor. In a point set compression framework, [DCV14] proposes a descriptor from which the surface can be locally resampled, based on a local surface height map. Following the success of the non-local means in image processing [BCM05], similarity based denoising has been proposed for surface meshes [YBS06] and point clouds [Dig12, GAB12]. Our method borrows the non-local processing idea from this thread of work. Several other descriptors gather different kind of informations and are particularly efficient for taking into account the shape at different scales [GBAL09]. However, a complete survey of such descriptors is beyond the scope of this paper since they are not well suited for resampling locally the surface from them. Self-similarities have also been used for addressing the inpainting problem [SACO04], which involves detecting similarities at a much larger scale than the one we are considering in this paper. Similarities can also be detected to extract global shape structures such as symmetries [MGP06]. This type of similarity is much more constrained (explicit symmetry) than the one we aim to explore here, at the level of very local surface patches. At a more global scale, Dekel et al. [DMIF15] analyze similarities, to correct or exaggerate discrepancies.

3. Overview

Our surface super-resolution approach can be summed up as follows. Let us write S the original scan and $\mathcal{S}$ the set of scans used to enhance S. If a single scan is used then $\mathcal{S} = \{S\}$, otherwise $\mathcal{S}$ contains all the scans used to enhance S (including S itself): $\mathcal{S} = \{S_1 = S, S_2, \cdots, S_n\}$. Our method is divided into four steps:

- For all points of all scans of $\mathcal{S}$, a descriptor based on a low resolution grid is computed representing the local surface scan around the point. Each descriptor encodes information of a single scan S_i.
- Secondly, the ℓ^2 distance is used to compare low-resolution descriptors. For every local descriptor computed at a point of S a similar patch query is performed among the descriptors of other points of $\mathcal{S}$.
- Similar descriptors are then aggregated locally, to produce high resolution descriptors all over the surface.
- The last step synthesizes a high resolution sampling over the surface, using regularization constraints during the merging of neighboring descriptors.

4. Local surface description

One of the key ingredient of our proposed super-resolution approach is a local surface descriptor that captures fine surface details. In the following, this descriptor will not only be used to characterize the surface locally but also to encode it, hence we will also refer to it as a patch. Most local surface explicit descriptions encode what lies above a local regression plane and consider therefore that surface details lie in any differential information with degree above 1. Yet this choice makes it difficult to detect similarities in highly curved surface areas. Furthermore, the similarity between descriptors is then governed by the degree 2 properties (such as the curvatures), and will in practice hardly take any finer details into account.

Our proposed descriptor aims at capturing the differential properties at such a scale where the sampling is dense enough and focuses on the residual part that corresponds to the details that may not be captured analytically. Indeed, our input surface can only be measured at a scale depending on the local density of the input points. We chose to describe each neighborhood by a height field over a quadric surface and perform the similarity search among the set of height fields. Thus, finer similarities can be found, as opposed to a description as a height field over a local plane as in [Dig12]

or [DCV14]. One could go further than a quadric surface, and consider a height field over a cubic surface for which efficient algorithms exist (e.g. [GI04]). However when increasing the order of the regression surface, more points - and consequently a larger neighborhood - are required to estimate it efficiently. A quadric is a good compromise between the capture of high order detail and locality of the description. Finally, our method can be easily adapted if need be to any smooth surface + height field description.

First, a quadric regression is performed around each point p in a radius r around p. A local coordinate frame is computed using principal component analysis on a local neighborhood. Expressing the neighboring points $(p_i)_{i=1\dots n}$ in this local coordinate system as (x_i, y_i, z_i), we look for a regression quadric Q such that $\forall i, z_i \simeq Q(x_i, y_i)$. If the chosen coordinate system is the intrinsic coordinate system given by the two principal directions and the normal, this quadric is expressed as $Q(x, y) = \frac{1}{2}k_1 x^2 + \frac{1}{2}k_2 y^2$. This formulation requires the computation of the principal directions $\mathbf{t}_1, \mathbf{t}_2$ and principal curvatures $k_1 \geq k_2$ of the surface at p. A way to achieve this is to compute a first regression quadric using an outlier-robust norm (ℓ^1 in our case), use it to estimate the first and second fundamental forms *I* and *II*, and derive the principal curvatures and directions. The ℓ^1 minimization is solved by an Iteratively Reweighted Least Squares process [BBS94]. Namely, to find Q minimizing $\sum_i \|z_i - Q(x_i, y_i)\|_1$, it introduces the weights w_i accounting for the likelihood for a point to be an inlier. w_i is initially set to 1, then, one iterates the two following steps: compute the weighted least squares regression and update the weights w_i depending on the distance to the regression quadric as $w_i = \frac{1}{\max(|z_i - Q(x_i, y_i)|, \varepsilon)}$, with ε a small numeric constant to avoid division by 0 (e.g. 10^{-5}). Thus, the quadric regression is less likely to drift in the presence of outliers.

Recall that our goal is to express the local surface as a height field over a quadric and that this height field should be discrete, to reflect the shape at a given scale. This is done in two steps: Anchor points are first sampled regularly on the regression quadric and serve as low resolution bin centers. At these anchor points, a height value is computed from neighboring input points. This yields a height field that will be used for similarity extraction.

In the first step, a grid is mapped onto a quadric surface so as to have equal bins in the sense that adjacent bins centers are geodesically equidistant. Each patch contains the exact same number of bins, which will be crucial for later comparisons. The geodesic width d_{lr} of bins being given, the bin centers are sampled as follows. Noticing that the quadric surface is driven by two curves $c_1(x) = k_1 x^2$ and $c_2(y) = k_2 y^2$, we compute the bin centers positions x_i, y_i such that the distance between $c_1(x_i)$ and $c_1(x_{i+1})$ (respectively $c_2(y_i)$ and $c_2(y_{i+1})$) over c_1 (resp. c_2) equals d_{lr}. With this definition, the geodesic size of the patch is the same for all patches but the ambient-space size depends on the local surface curvature. The captured variations are thus free of any lower-scale curvature information, which is inherently impossible with degree 1 descriptors.

The second step uses the obtained bin positions to compute the low-resolved input height field. The height of bin b is obtained by Gaussian interpolation of the height of the input points in a very small tangential neighborhood around b: the size of this neighborhood is comparable with the size d_{lr} of the bin. The local surface
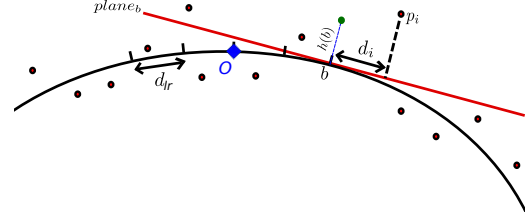
Figure 3: *Computation of the height of the bin $h(b)$ over the quadric surface using a local tangent plane at b, one of the bin of the patch. The center of the patch is marked with the blue diamond and named O.*

areas being thus discretized can now be compared even if they are differently sampled in the input scan. Ideally, this would involve computing the projection q_i of each point p_i on the quadric to obtain the height of p_i and the geodesic distance between q_i and each bin center. Such computation per patch bin and per point being prohibitive, we use an approximation instead. Let us consider a bin b of a patch $\mathcal{P}$, with normal n_b obtained in closed form as the normal to the quadric surface at the center of b. The height of bin b is computed as a weighted average of the heights of the neighboring points p_i projected on the plane $plane_b$ passing through the center of b with normal n_b (see Fig. 3). Thus:

$$h_{\mathcal{P}}(b) = \frac{\sum_{p_i} w(b, p_i) \cdot height(plane_b, p_i)}{\sum_{p_i} w(b, p_i)} \qquad (1)$$

where $w(b, p_i) = \exp\left(-\frac{\|p_i - b\|^2 - \langle p_i - b, n_b \rangle^2}{2d_{lr}^2}\right)$ is a Gaussian weight decreasing when the projection of p_i moves away from b. This height corresponds to the minimization of a ℓ^2 norm. To make the process more robust and alleviate the influence of outliers and noise, we can use a ℓ^1 norm instead and find the height $h_{\mathcal{P}}(b)$ as the minimizer h of $\sum_{p_i} w(b, p_i)|h - height(plane_b, p_i)|$. We compared this approximate height computation with a gradient descent to have a better estimation of the projection and our test experiments showed that the error on the projection is in practice negligible with respect to the computation time gain (less than 1% of the bin size in average).

Since point clouds often contain sampling irregularities, it may happen that the height of a bin cannot be computed because of a lack of input points nearby. In this case, the bin is classified as invalid and will not contribute to the later processing. Finally, each patch is supplemented with the location of the input points that have been used in its construction and that will later be used to generate the super-resolution.

5. Retrieving patch similarities

Once the patches are computed, they should be compared to detect the similar ones. To do so, a metric able to compare patches must be defined. In order to be identified as such, two similar patches should have matching height fields. Yet the orientation of the height fields depends on the local frame. Indeed, given the oriented normal and the principal directions, there remains an ambiguity on the principal

directions orientations. While two possible frames are possible, the choice between them needs to be consistent over the surface. To avoid having to choose between two orientations, we introduce two possible descriptors for each point, each one corresponding to a different orientation of the frame. Although it increases the total number of patches, it will barely affect the search for similarities. Indeed this search is done in a kd-tree and is therefore logarithmic w.r.t. the number of patches.

The patch description being a height field over a (distorted) grid, it can be interpreted as a vector of values in a high dimensional space $\mathbb{R}^{nbins_{lr}}$, $nbins_{lr}$ being the number of low-resolved bins. The similarity between patches can thus be computed using a slightly modified ℓ^2-norm:

$$d_{sim}(\mathcal{P}_a, \mathcal{P}_b) = \frac{\sum_{i=1}^{nbins_{lr}} \delta_i(\mathcal{P}_a, \mathcal{P}_b)|\mathcal{P}_a(i) - \mathcal{P}_b(i)|^2}{\sum_{i=1}^{nbins_{lr}} \delta_i(\mathcal{P}_a, \mathcal{P}_b)} \quad (2)$$

Where $\mathcal{P}_a(i)$ returns the height value in patch $\mathcal{P}_a$ of the i-th bin and $\delta_i(\mathcal{P}_a, \mathcal{P}_b)$ is an indicator function that is equal to 1 if the i-th bin is valid both in $\mathcal{P}_a$ and $\mathcal{P}_b$ and 0 otherwise. If both patches do not share enough valid bins this similarity distance is set to ∞. While the ℓ^2-norm is not flawless, it allows for a fast similarity search: using a kd-tree in $\mathbb{R}^{nbins_{lr}}$, looking for similar patches breaks down to a kd-tree query. The similarity distance yields a similarity weight stating how similar patches $\mathcal{P}_a$ and $\mathcal{P}_b$ are:

$$w_{sim}(\mathcal{P}_a, \mathcal{P}_b) = \exp\left(-\frac{d_{sim}(\mathcal{P}_a, \mathcal{P}_b)^2}{2r_{sim}^2}\right) \quad (3)$$

where r_{sim} is the radius for the similarity search.

Introducing the similarity weight allows to better take similarities into account, as patches that are very similar according to the comparison metric will have a weight close to 1 and will contribute more to the high resolution synthesis explained in the following section. r_{sim} is the distance threshold above which two patches are not considered as similar, so that there might remain areas that are not detected as similar to any other surface area.

Importantly enough, there will still be some patches that will be deemed not similar by our algorithm whereas they could be similar after some reorientation or repositioning of the patches. This can be due to the fact that they are centered at slightly different locations or that they are parameterized differently. Our method does not claim to find every instances of self similarity. However, missing similar patches will still yield a super-resolved point cloud, although it might not be as improved as it could have been if all similarities had been identified. As illustrated in figure 10, patches corresponding to a similar geometric residue over the quadric approximation are spread over the surface, which reinforces the hypothesis of self-similarity on the input point sets. It is also interesting to note that those similar details may be located in areas with different large scale curvatures.

6. Final Synthesis

The final surface synthesis involves two steps. First a super-resolved version of each patch is built by aggregating its similar patches. Then, super-resolved patches are consolidated and sampled to yield the final super-resolution surface.

6.1. Super-resolution patch synthesis

At this point of the algorithm, for each patch $\mathcal{P}_0$ we have a set D_0 of its similar patches along with their similarity weights. These similar patches can be seen as several acquisitions of the same surface detail. We are therefore in a classical super-resolution case, where several measures of the same signal are merged to get a better resolution signal, in our case a height field defined over a finer quadric grid. Recalling that each patch stores the set of points that were used in its construction, all the points associated to a similar patch of D_0 are aggregated and combined by Gaussian interpolation in the tangential neighborhood of a super-resolved bin b_{sr} (size $d_{sr} < d_{lr}$), yielding the height $\hat{h}$ at position b_{sr} as follows:

$$\hat{h}_{\mathcal{P}_0}(b_{sr}) = \frac{1}{C} \sum_{\mathcal{P} \in D_0} w_{sim(\mathcal{P}_0, \mathcal{P})} \sum_{p \in \mathcal{P}} w(b_{sr}, p) height(\mathcal{P}, p) \quad (4)$$

where $height(\mathcal{P}, p)$ is the distance of p to the quadric of $\mathcal{P}$, $w(b_{sr}, p) = \exp(-\frac{\|p - b_{sr}\|^2}{d_{sr}^2})$ and C a weight normalization constant $C = \sum_{\mathcal{P} \in D_0} w_{sim(\mathcal{P}_0, \mathcal{P})} \sum_{p \in \mathcal{P}} w(b_{sr}, p)$.

Once again the previous development assumes that the similar patches agree well in the overlapping areas at finer resolution. However, it is more realistic to assume that some patches might propose different behaviors for the super-resolved surface, even if the low-resolved patches match. To avoid such outliers interfering with the regression, once again we resort to a ℓ^1 norm minimization to recover the height of a super-resolved bin $\hat{h}_{\mathcal{P}_O}(b_{sr})$ as the minimizer h of:

$$\sum_{\mathcal{P} \in D_0} w_{sim}(\mathcal{P}_0, \mathcal{P}) \sum_{p \in \mathcal{P}} w(b_{sr}, p)|h - height(\mathcal{P}, p)| \quad (5)$$

Recall that, similarly to section 4, we want to avoid computing explicitly the projection of p on the quadric surface. Therefore, we use a similar approximation to compute $height(\mathcal{P}, p)$ depending on the considered bin b_{sr} of the finer grid mapped onto the quadrics : $height(\mathcal{P}, p) = \langle p - b_{sr}, n_{b_{sr}} \rangle$.

Equation 4 shows that in order to noticeably contribute to the height of b_{sr}, a point p must be close to the current bin b_{sr} and originate from a patch strongly similar to $\mathcal{P}_0$. Computing a finer height field differs from a simple interpolation since the original patch is actually enhanced by additional points carrying relevant information. Everything happens as if the quadrics of all the patches were deformed to fit the quadric of $\mathcal{P}_0$, and their associated input points were lifted at their corresponding heights. This process allows to superimpose local surface areas with different curvatures. If no similar patch is found, the synthesized high resolution patch is a plain interpolation of the input points attached to the patch. Thus our algorithm performs theoretically at least as well as a local surface interpolation.

6.2. Sampling the final surface

Once the super-resolution patches are computed, the final manifold surface $\mathcal{M}_{sr}$ must be sampled from this new information. We define a projection procedure that projects a point q in the ambient space onto the super-resolved surface using a partition of unity interpolation approach [OBA*03]. Recall that each input point p of the low-resolved input scans is the center of a patch that has been increased in resolution. The projection of a point q onto the super-resolved surface is computed as a linear combination of its projections on the super-resolved patches associated with the closest points in the input scans. In order to favor patches associated with the closest input points, the combination weights decrease exponentially with respect to the distance to q. This can be summed up as follows:

$$proj(q, \mathcal{M}_{sr}) = 1/C \sum_{p \in \mathcal{S}} proj(q, \mathcal{P}_p) w_{prox}(q, p) \qquad (6)$$

where $proj$ is the projection operator, $\mathcal{P}_p$ is the super-resolved patch associated to point p, $w_{prox}(q, p) = \frac{\exp(-\|p-q\|^2)}{2\sigma^2}$ and $C = \sum_{p \in \mathcal{S}} w_{prox}(q, p)$. Although the projection step involves an averaging of point positions, the averaging scale σ is comparable to the size of the super-resolved bins and avoids therefore any smoothing or degradation of the details.

However, it may happen that the super-resolved details introduced by neighboring patches do not fit. This is dealt with by introducing some robustness in the combination. Instead of using directly Equation 6, we find the projection as the minimizer $\tilde{q}$ of $\sum_{p \in \mathcal{S}} w_{prox}(p, \tilde{q}) \| proj(\tilde{q}, \mathcal{P}_p) - \tilde{q} \|_1$, which can be obtained via Iteratively Reweighted Least Squares and gives prominence to locally agreeing super-resolved patches.

Taking as input points in the ambient space all the positions given by all the super-resolved patches and projecting them onto $\mathcal{M}_{sr}$ as explained above yields a dense sampling of $\mathcal{M}_{sr}$. However the obtained density exceeds the obtained accuracy of the points. To get a more adequate sampling density with respect to the resolution gain, we perform a Poisson sampling step with a disk radius equal to d_{sr}, the width of the super-resolution bins.

7. Results and discussion

Our algorithm was implemented in C++ using Nanoflann for the high-dimensional kd-tree and the Eigen library for linear algebra computations. All patch computations as well as similar patches searches are performed in parallel using OpenMP. Our unoptimized C++ code takes 5 minutes to process an input point set of $100K$ points. We use either point-based rendering or a standard mesh reconstruction technique [KH13] for visualization purposes. For comparisons with MLS upsampling schemes, we generate a set of points lying very close to the surface, with the same cardinality as our super-resolution results, and project these points on the surface using either RIMLS [OGG09] or APSS [GG07],

Figures 1 and 4 illustrate the performances of our quadric based super-resolution algorithm, which we call QSR, on a single scan of a surface. One can see that repetitive textures are well enhanced by

our algorithm. Figure 5 compares our single-scan super-resolution with APSS and RIMLS interpolation methods as well as the super-resolution of Kil et al. [KMA06] using 100 scans. Kil et al. outperforms slightly our method but that is expected since it averages over a hundred scans, whereas our method works from a single scan. Both super-resolution methods restore more precise details as RIMLS and APSS upsampling. On Figure 6, a shape was smoothed out and downsampled, keeping only 17% of the initial points. Then this degraded version was upsampled using our super-resolution algorithm. One can see that, especially in the hair, the texture is enhanced while both RIMLS and APSS interpolations fail to enhance them, nor should they succeed since these are purely local schemes. However, one can observe that our super-resolution approach locally hallucinates features in local areas with high curvature such as the edge of the nostril. This highlights the importance of using a scale r well suited to the processed shape.

Figures 7 and 8 show the downsampling and degradation by additive noise of different variances of a high resolution shape and its recovery by our super-resolution process. Handling the added noise was only possible through the introduction of robust statistics in the estimation process. Although features that are clearly repeated such as the hair patterns are well recovered, less frequent details are lost. Table 1 compares our results with APSS and RIMLS as well as with non local means for point sets [Dig12] in terms of error w.r.t. the original. This error is computed as the root mean square error between the points of the initial full-resolution shape and their nearest neighbors in the resulting point clouds. Our algorithm clearly outperforms the other methods by taking into account the non local information. Note however that the bad performance of non local means can be expected since it is purely a denoising method that cannot be adapted for upsampling.

A more difficult challenge is set by the Braconnier dataset (Fig. 9) which corresponds to the head of a statue acquired by a mainstream acquisition device: a Microsoft Kinect. Devices of this kind produce very noisy depth images at a very hight rate. Each input scan is the average of 30 depth images taken from the exact same position, in order to reduce the input noise. One can see that an additional effect of our method is noise removal. The single-scan super-resolution indeed removes most of the noise, but tends to remove some well located details as well (around the eyes for example). On the contrary, our multiple scans super-resolution preserves the details better while still removing the noise. As a comparison, we also show two results of RIMLS interpolation for two different filter scales on a single scan. Both RIMLS results are outperformed by the super-resolution in the sense that they either enhance the noise and preserve the details or smooth both the noise and the details out.

Using several scans of the same object can be a clear advantage since their analysis yields a larger set of patches. Hence using the same similarity radius for the similar patches query will yield more similar patches than when using a single scan. In that case, it can be interesting to use a smaller similarity radius to get the same number of patches that would have been obtained with a single scan but with higher similarity (Figures 9, 10). On Figure 10, notice in particular how feather textures are well enhanced by the single-scan super-resolution and even better restored by the multi-scans super-
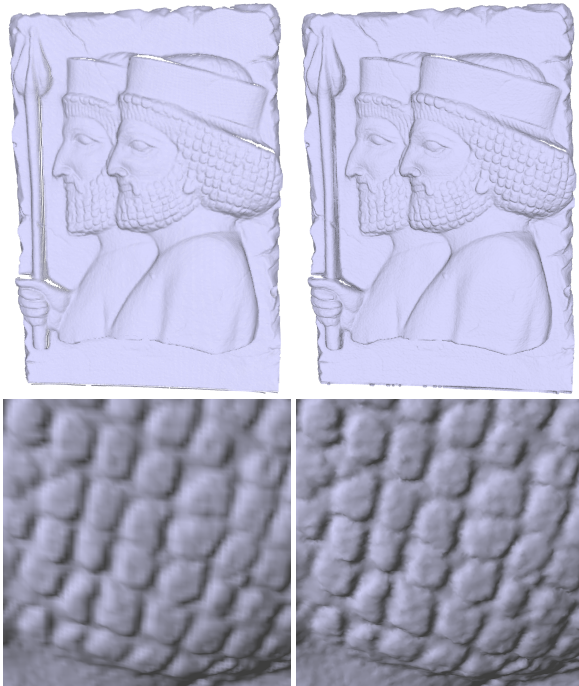
Figure 4: *Super-resolution of a single scan of the Persepolis point set. Left: initial scan, right: super-resolution. The shape details appear much sharper after the super-resolution process. Parameters: $r = 4$ (Shape diagonal: 114), $nbins_{lr} = 64$, $nbins_{sr} = 400$, $r_{sim} = 0.2$.*

Noise variance (% diagonal)	Initial error	QSR	APSS	RIMLS	NLMeans
0.01%	0.044	0.007	0.021	0.013	0.012
0.1%	0.053	0.019	0.049	0.048	0.039
0.5%	0.071	0.059	0.056	0.057	0.060

Table 1: *Comparison of the performance of our super-resolution algorithm QSR with other methods on the shape of Fig.7 : APSS [GG07], RIMLS [OGG09] and NLMeans [Dig12]. The test data is obtained by subsampling an initial point set (700K points) to 10% of its size, and adding a noise of variance equal to 0.01%, 0.1% and 0.5% of the diagonal.*

resolution. Figure 10 also shows the set of similar patches for a query patch located between two parrot feathers. Unsurprisingly the similar patches also correspond to such areas.

We now demonstrate empirically the advantage of building our descriptor over a quadric instead of a plane. Indeed the quadric+height descriptor is able to capture more similarities than its planar counterpart, because it frees itself from the larger scale curvature of the surface and concentrates on the smaller scale geometric texture. To illustrate this, we consider two sweep surfaces with similar repetitive details: a sinusoid mapped onto a plane and the same sinusoid mapped onto a cylinder (Figure 11). Although these shapes are different, their geometric textures are the same. The size of the patches was chosen so as to cover more than one pe-

Similarity radius	mean # similar patches (std. dev.)	
	plane+height	quadric+height
0.25	0.5(3.9)	57.6(60)
0.30	0.8(4.7)	181.7(114)
0.37	1.3(6.1)	428.1(151)
0.5	3(8.9)	766(152)
0.6	5(12.2)	877(105)
1	18(28.6)	934.7(28)

Table 2: *Average number of similar patches found by querying a patch from the cylinder-based shape (see Fig 11) in the patches extracted from the planar-based shape and vice versa, using various similarity radii. Much more similar patches are found using a quadric+height descriptor. The width of the patch is set to 1.5 sinusoid period.*

riod of the sinusoid, so that the sinusoid is included in the geometric texture. Then, for each patch of the cylinder-based shape (respectively the plane-based shape), we look for its similar patches in the set of patches from the plane-based shape (resp. the cylinder-based shape). The average number of similar patches found are reported in table 2. As expected, this number is higher for the quadric+height descriptor for all similarity radius values. The quadric+height descriptor here encodes the phase shift between the sinusoids. On the contrary, the number of similar patches is low for the plane+height descriptor since the difference of height over the tangent plane encodes mostly the difference of curvature. Hence, the first shape will not be used to increase the resolution of the second with the plane+height descriptor.

7.1. Parameters

As described above, our algorithm depends on 4 parameters, which should be tuned carefully depending on the geometry of the shape, and the precision of its sampling. These parameters are the patch radius r, the grid size for both low $nbins_{lr}$ and super-resolution $nbins_{sr}$ grids and the similarity radius r_{sim} for the similar query search. In practice, only two parameters govern our whole algorithm: the patch radius and the similarity radius r_{sim}, since the other parameters can be deduced from them. The patch radius size is related to the size of the details we want to capture, if it is too small it will fail to capture interesting details, conversely if it is too large, few similar patches will be found. The number of low-resolved bins $nbins_{lr}$ stems directly from the patch radius. We set it equal to the mean number of points in a neighborhood of radius equal to the patch radius, so that the neighborhood will contain as many points as there are low-resolved bins in average. Using r_{sim}, one can compute the average number of similar patches and the sum of their similarity weights. This sum gives an estimate of the mean number K of similar patches that can be obtained in practice. Then the size of the super-resolution grid can be expressed as $\sqrt{K nbins_{lr}}$. Those points are then consolidated by merging points that fall below the super-resolution bin size. In this setting, the expected resolution gain is then $K nbins_{lr}$.
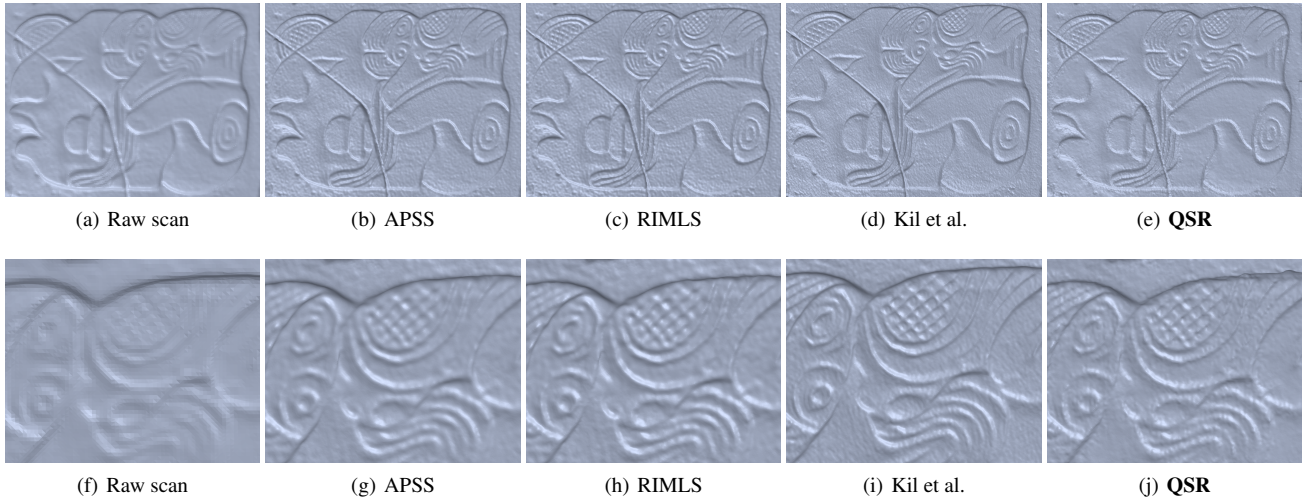
Figure 5: *Comparisons of our single-scan method with APSS interpolation [GG07], RIMLS interpolation [OGG09] and Kil et al. [KMA06] using* 100 *scans (meshes obtained by Poisson Reconstruction). Close-ups on a detailed area are shown on the bottom row. Parameters:* $r = 4$ *(Shape diagonal:* 323), $nbins_{lr} = 64$, $nbins_{sr} = 400$, $r_{sim} = 0.2$.
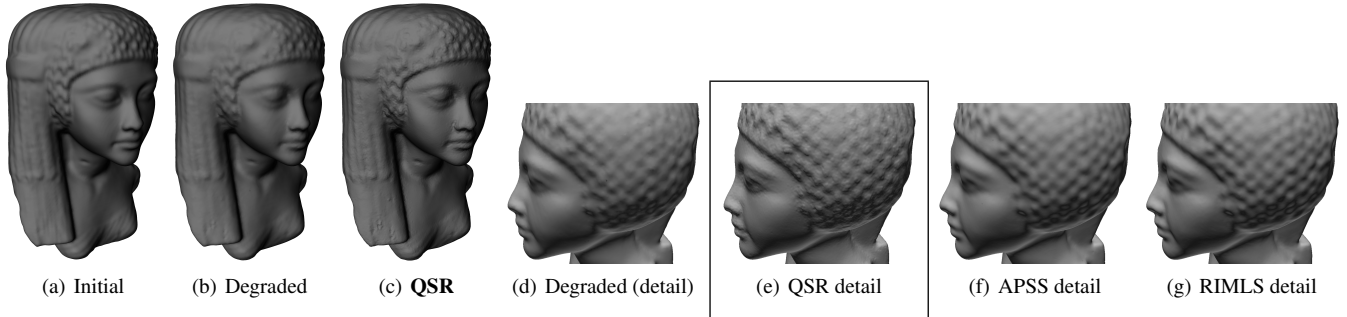
Figure 6: *Comparisons on a whole object surface. Original point set, smoothed out and downsampled shape, super-resolution result QSR, APSS and RIMLS interpolations. Bottom row: degraded shape, super-resolution result, APSS and RIMLS interpolations (meshes obtained by Poisson Reconstruction). Parameters:* $r = 4$ *(Shape Diagonal:* 222.05), $nbins_{lr} = 25$, $nbins_{sr} = 100$, $r_{sim} = 0.05$.

## 7.2. Limitations and failure cases

The principal limitation of our approach stems from the self-similarity hypothesis. As explained in the introduction many surfaces are self-similar to some extent. However, if the surface does not have this property, it is difficult to extract similar patches and an interpolation is performed. Thus the choice of the similarity radius is very important in this particular case. Furthermore, our algorithm tends to hallucinate features near high-curvature edges, where the radius patch becomes to large, which leads to errors in the quadric approximation. Although we handled noise with robust norms, outliers might still cause some bad estimates, since a patch centered at an outlier will have a lot of invalid bins. To alleviate this effect, a preprocessing step filters out outliers based on some local density criterion. Furthermore the resilience to noise holds only for relatively small amounts of noise. Another limitation is the difficulty to choose the parameters to run the algorithm. Although we gave some heuristic rules to set the parameters in the previous paragraph, an automatic way of setting the parameters remains to be designed.

A failure case of our super-resolved patch generation is when some patches are similar at the lower scale of the analysis but have inverted details at a higher scale. Such would be the case of sinusoids with locally inverted phase mapped onto quadric surfaces. At intersecting points of the sinusoidal residues, and if the bin size of the low resolution grid coincides with half the wavelength of the sinusoids, the resulting super-resolved patch would locally remove the details, and be almost equal to the quadric surface. However, such a behavior can only be local, and corresponds to a singular case with specific sampling conditions unlikely to happen in practice. Furthermore, the variety of patches and robust synthesis of the surface limit the impact of such local behaviors on the final result.

Conclusion

We introduced a new method for enhancing the resolution of a 3D scan and more generally the precision of an unstructured consolidated point set. The approach compares local surface patches at a level of detail that exceeds the order 2 differential properties,
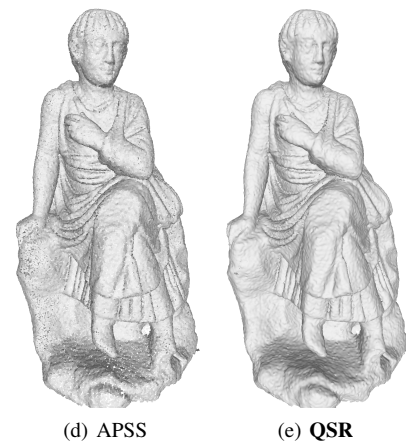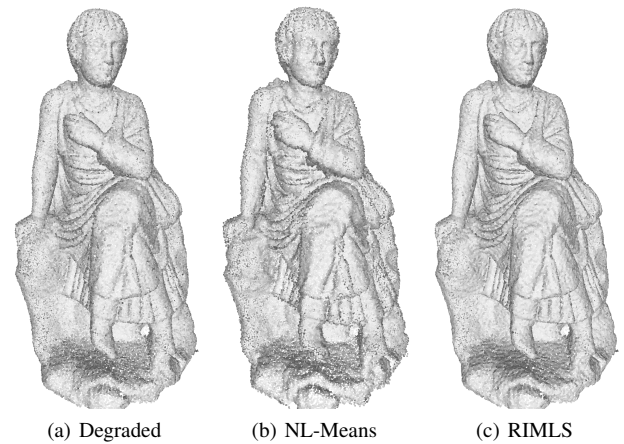
(a) Initial   (b) Degraded   (c) RIMLS   (d) **QSR**
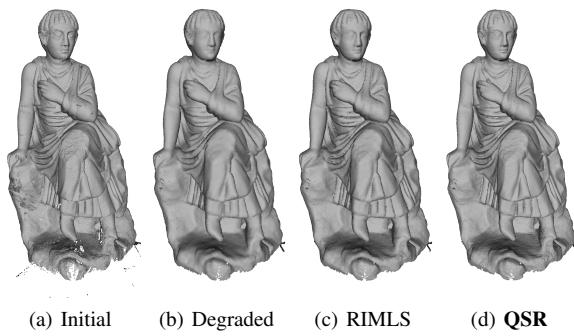
Figure 7: *Groundtruth Comparison: super-resolution of an artificially degraded shape. The degradation involves a subsampling to* 10% *of the points and an additional Gaussian noise of variance* 0.001% *of the diagonal. Processing radius: $r = 0.2$ (Shape diagonal: 23.83), $nbins_{lr} = 25$, $nbins_{sr} = 100$, $r_{sim} = 0.05$. Normals are computed using PCA with* 10 *neighbors for visualization purpose. Although our method successfully enhances some lost features (hair patterns), it fails to recover features that are not frequent enough (eyes).*

building super-resolved patches and aggregating the result in a high resolution surface. We demonstrated that our method was able to perform single-scan super-resolution but also to generate a high resolution surface from several scans without needing any costly registration step. The local descriptor we introduced can be used in various applications, one of those being nonrigid deformation, an idea we want to explore in a future work. From a more practical point of view, the computation times are unfortunately still not small enough for real-time applications, someting we want to improve in a future work.
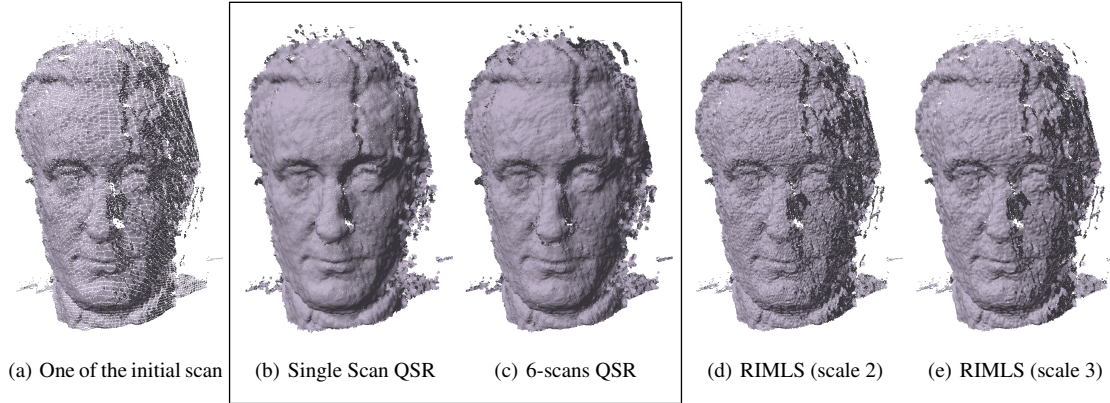
References

[AKSA09] ABBASINEJAD F., KIL Y. J., SHARF A., AMENTA N.: Rotating scans for systematic error removal. In *Proc. SGP* (2009), pp. 1319–1326. 2

[BBS94] BURRUS C. S., BARRETO J. A., SELESNICK I. W.: Iterative reweighted least-squares design of fir filters. *IEEE Transactions on Signal Processing 42*, 11 (Nov 1994), 2926–2936. 4

[BCM05] BUADES A., COLL B., MOREL J. M.: A non-local algorithm for image denoising. In *Proc. IEEE CVPR 2005* (2005), vol. 2, pp. 60–65. 3

[BK02] BAKER S., KANADE T.: Limits on super-resolution and how to break them. *IEEE Trans. PAMI 24*, 9 (2002), 1167–1183. 2

[DCV14] DIGNE J., CHAINE R., VALETTE S.: Self-similarity for accurate compression of point sampled surfaces. *Computer Graphics Forum 33*, 2 (2014), 155–164. Proc. Eurographics 2014. 3, 4

[Dig12] DIGNE J.: Similarity based filtering of point clouds. In *CVPR Workshops* (2012), IEEE, pp. 73–79. 3, 6, 7

(a) Degraded   (b) NL-Means   (c) RIMLS

(d) APSS   (e) **QSR**

Figure 8: *Groundtruth Comparison: super-resolution of an artificially degraded shape. The degradation involves a subsampling to* 10% *of the points and an additional Gaussian noise of variance* 0.005% *of the diagonal. Processing radius: $r = 0.2$ (Shape diagonal: 23.83), $nbins_{lr} = 25$, $nbins_{sr} = 100$, $r_{sim} = 0.05$. Normals are computed using PCA with* 10 *neighbors for visualization purpose.*

[DMIF15] DEKEL T., MICHAELI T., IRANI M., FREEMAN W. T.: Revealing and modifying non-local variations in a single image. *ACM Trans. Graph. (SIGGRAPH Asia)* (2015). 3

[EF97] ELAD M., FEUER A.: Restoration of a single superresolution image from several blurred, noisy, and undersampled measured images. *Trans. Img. Proc. 6*, 12 (1997), 1646–1658. 2

[EHO01] ELAD M., HEL-OR Y.: A fast super-resolution reconstruction algorithm for pure translational motion and common space-invariant blur. *IEEE Trans. Image Processing 10*, 8 (2001), 1187–1193. 2

[EV07] EBRAHIMI M., VRSCAY E. R.: Solving the inverse problem of image zooming using "self-examples". In *Proc. of the 4th Int. Conf. on Image Analysis and Recognition* (2007), ICIAR'07, pp. 117–130. 2

[FF11] FREEDMAN G., FATTAL R.: Image and video upscaling from local self-examples. *ACM Trans. Graph. 30*, 2 (2011), 12:1–12:11. 2

[FJP02] FREEMAN W. T., JONES T. R., PASZTOR E. C.: Example-based super-resolution. *IEEE Comput. Graph. Appl. 22*, 2 (2002), 56–65. 2

[GAB12] GUILLEMOT T., ALMANSA A., BOUBEKEUR T.: Non local point set surfaces. In *Proc. 3DIMPVT 2012* (2012). 3

(a) One of the initial scan   (b) Single Scan QSR   (c) 6-scans QSR   (d) RIMLS (scale 2)   (e) RIMLS (scale 3)

Figure 9: *Super-resolution on a Kinect scan of a statue head. Top row: From left to right: initial scan, result of QSR using a single scan, result of QSR using 6 different scans of the same object, results of RIMLS resampling using scales 2 and 3. Parameters: $r = 2$ (Shape diagonal 1139), $nbins_{lr} = 16$, $nbins_{sr} = 64$, $r_{sim} = 0.05$.*
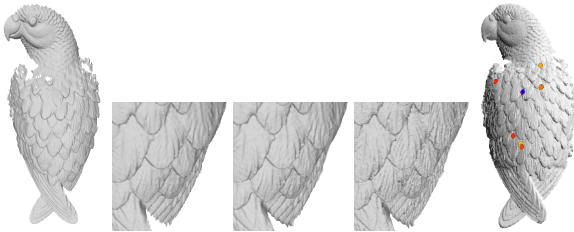
Figure 10: *Parrot super-resolution. From left to right: original scan, original scan detail, single-scan QSR detail, 4-scans QSR detail, 10 most similar patches found for the patch shown with a blue diamond. The color scale shows the proximity to the query patch. Parameters: $r = 4$ (Shape diagonal: 158), $nbins_{lr} = 64$, $nbins_{sr} = 400$, $r_{sim} = 0.2$.*
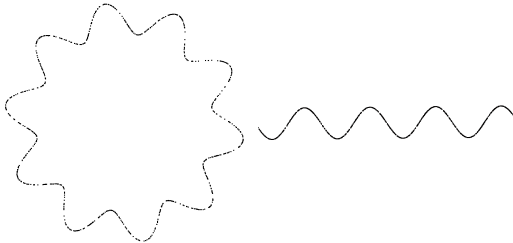
Figure 11: *Two similar synthetic sweep surfaces : A sinusoidal component was added in the normal direction. Similar patch queries are reported in table 2.*

[GBAL09] GEBAL K., BAERENTZEN J. A., AANAES H., LARSEN R.: Shape analysis using the auto diffusion function. *Computer Graphics Forum 28*, 5 (2009), 1405–1413. 3

[GBI09] GLASNER D., BAGON S., IRANI M.: Super-resolution from a single image. In *ICCV* (2009). 1, 2

[GG07] GUENNEBAUD G., GROSS M.: Algebraic point set surfaces. *ACM Trans. Graph. 26* (2007). Proc. SIGGRAPH 2007. 2, 6, 7, 8

[GI04] GOLDFEATHER J., INTERRANTE V.: A novel cubic-order algorithm for approximating principal direction vectors. *ACM Trans. Graph. 23*, 1 (2004), 45–63. 4

[GTB14] GUY E., THIERY J.-M., BOUBEKEUR T.: Simselect: similarity-based selection for 3d surfaces. *Computer Graphics Forum (Proc. EUROGRAPHICS 2014) 33*, 2 (2014), 165–173. 3

[JH99] JOHNSON A. E., HEBERT M.: Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE PAMI 21* (1999), 433–449. 3

[KH13] KAZHDAN M., HOPPE H.: Screened poisson-surface reconstruction. *ACM Trans. Graph.* (2013). 6

[KMA06] KIL Y. J., MEDEROS B., AMENTA N.: Laser scanner super-resolution. In *Proc. Point-Based Graphics* (2006), SPBG'06, pp. 9–16. 2, 6, 8

[MGP06] MITRA N. J., GUIBAS L., PAULY M.: Partial and approximate symmetry detection for 3d geometry. *ACM Trans. Graph. (SIGGRAPH) 25*, 3 (2006), 560–568. 3

[OBA*03] OHTAKE Y., BELYAEV A., ALEXA M., TURK G., SEIDEL H.-P.: Multi-level partition of unity implicits. *ACM Trans. Graph. 22*, 3 (2003), 463–470. 6

[OGG09] OZTIRELI A. C., GUENNEBAUD G., GROSS M.: Feature preserving point set surfaces based on non-linear kernel regression. *CGF 28* (2009), 493–501(9). 2, 6, 7, 8

[PE14] PELEG T., ELAD M.: A statistical prediction model based on sparse representations for single image super-resolution. *IEEE Trans. Image Processing 23*, 6 (2014), 2569–2582. 2

[PH93] PENTLAND A., HOROWITZ B.: Digital images and human vision. 1993, ch. A Practical Approach to Fractal-based Image Compression, pp. 53–59. 2

[SA01] SUN Y., ABIDI M. A.: Surface matching by 3d point's fingerprint. In *Proc. ICCV 2001* (2001), vol. 2, pp. 263–269. 3

[SACO04] SHARF A., ALEXA M., COHEN-OR D.: Context-based surface completion. *ACM Trans. Graph. 23*, 3 (2004), 878–887. 3

[SSXS08] SUN J., SUN J., XU Z., SHUM H.-Y.: Image super-resolution using gradient profile prior. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on* (2008), pp. 1–8. 2

[YBS06] YOSHIZAWA S., BELYAEV A., SEIDEL H.-P.: Smoothing by example: Mesh denoising by averaging with similarity-based weights. In *SMI '06* (2006), IEEE, p. 9. 3

[YWHM10] YANG J., WRIGHT J., HUANG T., MA Y.: Image super-resolution via sparse representation. *IEEE Trans. Image Processing 19*, 11 (2010), 2861–2873. 2