

A Novel 3D Vorticity based Approach for Automatic Registration of Low Resolution Range Images

Syed Afaq Ali Shah^{a,*}, Mohammed Bennamoun^a, Farid Boussaid^b

^a*School of Computer Science and Software Engineering,*

^b*School of Electrical, Electronics and Computer Engineering, The University of Western Australia, Perth 6009, Australia*

Abstract

This paper tackles the problem of feature matching and range image registration. Our approach is based on a novel set of discriminating three-dimensional (3D) local features, named 3D-Vor (**V**orticity). In contrast to conventional local feature representation techniques, which use the vector field (i.e. surface normals) to just construct their local reference frames, the proposed feature representation exploits the vorticity of the vector field computed at each point of the local surface to capture the distinctive characteristics at each point of the underlying 3D surface. The 3D-Vor descriptors of two range images are then matched using a fully automatic feature matching algorithm which identifies correspondences between the two range images. Correspondences are verified in a local validation step of the proposed algorithm and used for the pairwise registration of the range images. Quantitative results on low resolution Kinect 3D data (Washington RGB-D dataset) show that our proposed automatic registration algorithm is accurate and computationally efficient. The performance evaluation of the proposed descriptor was also carried out on the challenging low resolution Washington RGB-D (Kinect) object dataset, for the tasks of automatic range image registration. Reported experimental results show that the proposed local surface descriptor is robust to resolution, noise and more accurate than state-of-the-art techniques. It achieves 90% registration accuracy compared to 50%, 69.2% and 52% for spin image, 3D SURF and SISI/LD-SIFT descriptors, respectively.

Keywords: Automatic registration, Vector field, Vorticity, Range images.

1. Introduction

Feature representation and correspondence are active research areas in computer vision with numerous applications including reverse engineering, 3D modeling, 3D object recognition, augmented reality and medical diagnosis [1, 2, 3]. The aim of feature representation is to encapsulate the predominant information of the underlying

*Corresponding author

Email address: afaq@csse.uwa.edu.au (Syed Afaq Ali Shah)

local surface to provide sufficient descriptiveness. Feature correspondence, on the other hand, computes the similarity between the surfaces.

A variety of local feature representation techniques can be found in the recent literature. Keller and Darom [4] proposed scale invariant spin image (SISI) and local depth SIFT (LD-SIFT) descriptors for 3D mesh models. In their proposed technique, the SISI descriptor is constructed by computing the spin image descriptor over a local scale, while LD-SIFT is computed by representing the vicinity of the feature points as a depth map. Knopp et al. [5] proposed the 3D SURF descriptor, which is an extension of 2D SURF. The 3D object is voxelized in a volumetric cube using the intersection of faces with the grid-bins. The 3D SURF descriptor is then computed at the maxima of the voxelized grid. Most existing feature descriptor techniques have mainly been geared towards two dimensional (2D) data or high resolution 3D data captured using relatively expensive 3D sensors [4, 6, 7, 8, 9, 10]. These techniques suffer from either low descriptiveness or weak robustness when used with low resolution data (e.g., Kinect data) [11]. To address the problems of distinctiveness and robustness to low resolution data, we proposed the 3D-Div descriptor in [12]. Based on vector field analysis, the proposed descriptor exploits the divergence of the vector field for local surface description. Divergence measures the spreading out and the contraction of the vector field at each point of the surface. The values of divergence computed at each point of the surface are concatenated to form the 3D-Div descriptor. The latter has been shown to achieve good performance on low resolution data [12, 13]. The robustness to mesh resolution is achieved by using all the points in the local surface patch. Being a differential invariant of curves and surfaces, the divergence of a vector field captures the most significant information about surface variations at each point. It is however, sensitive to noise.

In this paper, we tackle the noise sensitivity which is inherent to the computation of differentials (and its invariants like divergence). We introduce a novel descriptor for low resolution data that exploits the vorticity of the vector field to capture the distinctive surface characteristics at each point of the local neighborhood to generate the proposed descriptor, referred to as 3D-Vor. In contrast to [12], we explore the tangential nature of the vector field (circulation in this case) and integrate the vector field over a small support radius to derive the proposed robust and distinctive local feature 3D-Vor. The integration operation carried out during the computation of the vorticity has a smoothing effect, providing increased robustness to noise, while alleviating the need for pre-processing. In addition, the locally tangent circulation of the vector field results in high signal-to-noise ratio and thus robust feature representation.

As robust feature representation of 3D surfaces, feature correspondence is also a challenging computer vision problem. The computation of the similarity between 3D surfaces is a critical task [13, 14], as the accuracy of coarse registration significantly relies on an accurate feature correspondence. Depending on the feature correspondence, coarse registration can be performed either manually or automatically. In the manual approach, corresponding points are manually selected between the overlapping regions of a pair of surfaces. These correspondences are then used to derive a rigid transformation (rotation and translation) that aligns the surfaces. On the other hand, in automatic coarse registration, the feature matching algorithm automatically identifies the corresponding points between the two surfaces and coarsely registers them by minimizing

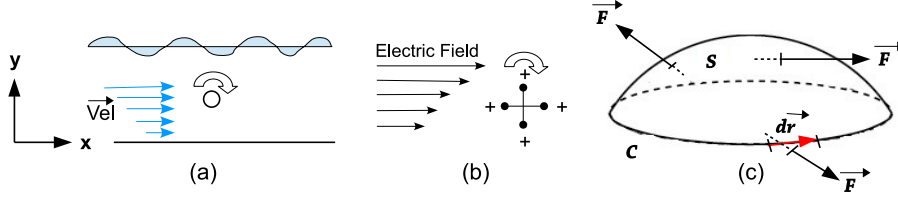


Figure 1: Illustration of the concept of vorticity (a) in the case of a velocity vector field \vec{V}_{el} , (b) the electric field which produces a force on a charge, (c) a vector field \vec{F} across a 3D surface.

the distance between the points [1].

The idea behind feature correspondence based automatic coarse registration is to represent the features of each range image and match these representations in order to identify corresponding points [1]. 3D object registration based on feature matching has been extensively researched during the last two decades [15, 16]. However, automatic registration of low resolution noisy range images is a challenging problem that has received no attention. In this paper, we propose a fully automatic feature correspondence and pairwise registration algorithm. Our algorithm uses a novel vorticity-based representation (3D-Vor) to represent local patches of a range image of a 3D object. 3D-Vors of two overlapping views are then matched to establish pairwise correspondences between the views. Corresponding 3D-Vor descriptors are verified in the local validation step of the proposed registration algorithm and then used to pairwise register the views. A pairwise registration is accepted only if it passes a local validation stage. Otherwise, it is rejected and another pair of matching 3D-Vor descriptors is sought. In the following section, we describe the novel concept of vector field's vorticity for feature representation in computer vision.

1.1. Intuitive Interpretation of Vorticity

To provide an intuitive interpretation of the concept of vector field's vorticity, we consider two examples. Suppose a vector field describing the velocity field of a fluid flow in a channel, with a paddle wheel placed in the liquid (Fig. 1(a)). The small solid arrows represent the velocity vectors of the fluid \vec{V}_{el} , being zero at the boundary and increasing upward. The top of the paddle wheel will have a greater force on it than the bottom due to the different fluid velocities. This will cause a torque on the paddle wheel, which is regarded as a vector along the paddle wheel axis. This torque is normal to the velocity vector, and in this case is directed (using the right-hand rule) into the page. The velocity vector field is said to have vorticity, which is represented by the torque normal to the vector field.

Let's consider another example, suppose we have an electric field with the spatial distribution shown in Fig.1(b). We construct a very small electrostatic paddle wheel consisting of two crossed insulating bars with small positive charges at each end. From Fig. 1(b), it can be seen that the topmost charge will have the largest force so that the paddle wheel will rotate as indicated. The rotation will be represented by the normal to and into the plane of Fig. 1(b).

These concepts of fluid mechanics and electric field theories are also applicable to 3D surfaces. Suppose we have a closed-line path C that is the contour of the surface S

(Fig. 1(c)). Also, assume there is a vector field¹ \vec{F} which passes through the surface. Let $d\vec{r}$ be an increment of the contour C . Then according to Stoke's theorem, the vorticity of the vector field \vec{F} can be represented by:

$$vorticity = \lim_{r \rightarrow 0} \frac{1}{|A|} \oint_C \vec{F} \cdot d\vec{r} \quad (1)$$

In Eq. 1, $\oint_C \vec{F} \cdot d\vec{r}$ is the component of the vector around the contour of the 3D surface. It indicates how much the vector field \vec{F} tends to circulate around the curve C . This line integral, derived from Stoke's theorem, is a generic expression which can be applied to non-rigid (e.g., fluids) or rigid (e.g., surfaces) objects for the analysis of fields in various domains. For example, \vec{F} in the integral part of Eq. 1, could represent the velocity vector for the analysis of fluid flow, wind force in the case of wind analysis, electric field force in the case of a charged body or vector field in the case of 3D surfaces. The concept of vorticity is therefore not restricted to fluid mechanics only, but it can be equally applied to rigid objects.

The vorticity of the vector field has also been shown to relate to 3D surfaces, by Longuet-Higgins [18]. For a concave surface, the curvature is negative. The circulation of the vector field is in the counter clockwise direction and resulting vorticity over the surface is positive, while the opposite applies in the case of convex surfaces. Based on the explanation of vorticity given by Longuet-Higgins, vorticity can capture the intrinsic curvature at a given point on the surface. We exploit these facts to capture the local surface information. We extract the local surface patch around each detected feature point. We next compute the vector field at each point of the local surface patch and calculate the vorticity of the vector field, using Eq. 1. The locally tangent circulation of the vector field results in a high signal-to-noise ratio and thus robust features. In addition, the integration operation carried out in the computation of vorticity has a smoothing effect, providing increased robustness to noise.

The vorticity is a point function², with different values, at different points in the vector field. The vorticity values computed at each point of the local surface are concatenated to construct the proposed 3D-Vor descriptor. The robustness to mesh resolution is achieved by using all the points of the 3D local surface for feature representation. As demonstrated in Sections 3 and 5, the proposed descriptor, which represents the vorticity of the field at each point, captures the distinctive characteristics of the underlying 3D local surface. Since the focus here is on the design of a novel local feature descriptor and its application to 3D surface modeling, the details about the detection of the feature points and their repeatability is outside the scope of this paper and is therefore omitted.

¹A vector field is a vector associated with every point on the 3D surface [17]. The vector of a point (vertex) is defined as the weighted average of the normals of its immediate neighbouring triangles [1].

²A point function $k = f(P)$ is a function that assigns some number or value k to each point P of some region \mathbb{R} of space

1.2. Paper Contributions

The contributions of this paper can be summarized as follows:

- A novel local surface descriptor 3D-Vor exploiting the vorticity of the vector field at each point of the local surface to capture distinctive and robust features of the underlying 3D surface.
- Besides a novel feature descriptor, a fully automatic feature matching algorithm for coarse registration of range images is also presented. The 3D-Vors of two overlapping views are matched to establish pairwise correspondences between the views. The 3D-Vor descriptors are verified in the local validation step of the proposed algorithm and then used to pairwise register the views.

1.3. Organization of the paper

The rest of this paper is organized as follows. The next section presents related work. Section 3 introduces the proposed local surface descriptor, namely 3D-Vor. Section 4 gives details of our automatic pairwise registration algorithm. Experimental results together with a detailed quantitative/qualitative analysis are provided in Section 5. Section 6 presents a performance comparison with state-of-the-art feature representation techniques. In section 7, we briefly describe multiview range image registration and provide qualitative results. Finally, a conclusion is given in Section 8.

2. Related Work

Various surface representation techniques have been used for feature matching and automatic coarse registration. The following is a brief historical survey of the related work in the area of automatic coarse registration by feature correspondences. Stamos and Leordeanu [19] proposed a straight-line based method to find pairs of correspondences between two views. Their technique is based on the extraction of straight-line segments directly from the range images, which are further registered with the aim of computing the transformations between the different views [20]. Chen et al. [21] proposed the renowned RANSAC-based DARCES algorithm which is based upon an exhaustive search. However, the latter is not practical for large datasets. Moreover, the DARCES algorithm makes some unrealistic assumptions about the overlapping regions of the views [20]. An approach to estimate the rotation between two surfaces represented as polynomial models has been proposed by Tarel et al. [22]. The method is based on obtaining a function of the distance between the polynomial model and the points, where these distances are nearly zero. The principal drawback of this method is the requirement that a large part of both images must belong to the overlapping region. Moreover, it is also sensitive to variations in mesh resolution [20]. Brunnstrom and Stoddart [23] proposed a genetic algorithm to find the correspondences between two range images. In their proposed technique, they define a vector containing the n index of correspondences between both range images. They set the size of the vector to n , i.e. the number of points in the second range image. The shortcoming of their technique is that the computation time is expensive, especially in the presence of a large number of points [20]. Wyngaerd et al. [24] proposed bitangent curve matching, based on first

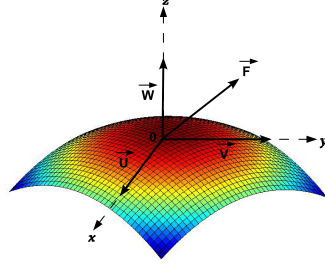


Figure 2: Vector Field. The three components of the vector field ($U(x,y,z)$, $V(x,y,z)$, $W(x,y,z)$) are shown along the x , y and z directions respectively.

order derivatives, making it therefore sensitive to noise [20]. Another limitation of bi-tangent curves is that they represent global features, which may not be contained fully inside the overlapping region of the views. The Spherical Attribute Imaging (SAI), proposed by Higuchi et al. [25], requires the underlying 3D surfaces to be free of topological holes, which limits its applicability. The feature matching technique proposed by Roth [26] is limited by the fact that it significantly relies upon the amount of texture on the object surface for consistent extraction of features. Unlike most existing registration techniques, the local surface descriptor based range image registration, described in this paper, is robust to variations in mesh resolution and noise, enabling it to achieve highly accurate registration on low resolution 3D data (Section 5).

3. Local Surface Description

In this section, we describe the different steps required to construct the proposed novel rotation invariant local surface descriptor called 3D-Vor.

3.1. Vector Field

Interpreted as optical flow fields, velocities of fluid particles or directions of strokes in a painting, vector fields are ubiquitous entities in computer vision, graphics and engineering [17]. The classical theory of physics is built upon the characterization of vector fields induced by the motion of an object. This fact is at the origin of the pervasiveness of vector fields in graphics applications, where they often appear as data from physical simulation or measurements of some real phenomenon [17]. Motivated by this, we propose to exploit the vector field for local feature description so as to better capture the geometrical information of the underlying 3D surface.

To compute the vector field, the input range images in the form of pointclouds, are converted into triangular meshes. This is performed by mapping the 3D points onto the 2D retinal plane of the sensor and performing a 2D Delaunay triangulation over the mapped points, as done in [1]. After triangulation, the points are mapped back to the 3D space and the triangles with edges longer than a prespecified threshold are removed. This separates surfaces falsely connected by the Delaunay triangulation. In our implementation, we removed all triangles with an edge length 0.6 standard deviations longer than the mean. For a given mesh, we define a small circular patch (local surface mesh) γ of radius r centred at the feature point P_k [4], taken as the origin of the x - y

Table 1: Number of feature points detected on five 3D objects with different shapes and pointcloud size.

Object	Pointcloud Resolution	Feature Points Detected
Bowl	10600	384
Binder	19100	286
Cap	16340	321
Cereal Box	37300	863
Food Box	7799	89

plane (Fig. 3(a)). To determine the feature points P_k , we used the well-known DoG feature point detector recently proposed in [4]. The DoG detector was chosen because of its low complexity and high repeatability. Using DoG detector [4], feature points P_k are detected automatically. There is no relationship between the number of feature points and the support radius r . However, the number of feature points depends on the size and shape of the 3D pointcloud. Table 1 reports the feature points detected for 3D objects with different pointcloud resolution. Since the focus here is on the design of a novel local feature descriptor, any further details about the detection of feature points and their repeatability is outside of the scope of this paper. Note that any other feature point detection method could have also been adopted [27]. In Section 6, we compared the registration performance of DoG with a more complex state-of-the-art feature point detector [27].

The vector field at each point (vertex) of the 3D surface are defined as the weighted average of the normals of its immediate neighboring triangles. The vector field can be represented as:

$$\vec{F}(x, y, z) = (U(x, y, z), V(x, y, z), W(x, y, z)) = \bar{U}\hat{x} + \bar{V}\hat{y} + \bar{W}\hat{z} \quad (2)$$

In Eq. 2, $(U(x, y, z), V(x, y, z), W(x, y, z))$ are the three components of the vector field in the x , y and z directions respectively, as shown in Fig. 2. Similarly, \hat{x} , \hat{y} and \hat{z} are the unit vectors in the x , y and z directions respectively. The vector field for a local surface patch γ is shown in Fig. 3(b). As it is difficult for a 3D sensor to capture the whole object through a single acquisition, the different views of the 3D object need to be acquired from different viewpoints. Consequently, there will be some rotation and translation between these 3D views (assuming rigid objects). For an accurate registration, it is mandatory that the local surface descriptor is invariant to rotation and translation. To achieve this, we construct a unique local coordinate system using the technique proposed in [11]. Next, the vector field \vec{F} for the local surface γ is aligned with the vectors of the local coordinate system to construct the rotation invariant local surface descriptor described, in the following section.

3.2. 3D-Vor Descriptor

The 3D-Vor descriptor for the entire local surface γ is computed as follows. Since we assume the circular patch to be small and \vec{F} differentiable, we take the first order approximation of the vector field \vec{F} at the point P_k i.e. the origin $0(0, 0, 0)$:

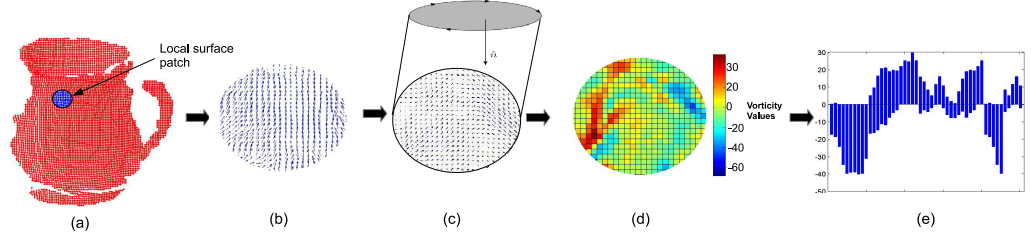


Figure 3: An illustration of the construction of a 3D-Vor feature descriptor. (a) The coffee mug partial view and the local surface (shown in blue) around a feature point. (b) Vector field computed for all the points of the local surface. (c) Circulation of the vector field is shown for the circular patch by small arrows over the local surface. (d) Vorticity values (i.e. the attributes of the resulting vector \hat{n} computed using the right hand rule) displayed for each point of the local surface patch. (e) The vorticity values are concatenated to form a 3D-Vor descriptor (Eq. 15). This Figure is best viewed in color.

$$U(x, y, z) = U(0) + U_x(0)x + U_y(0)y + U_z(0)z \quad (3)$$

$$V(x, y, z) = V(0) + V_x(0)x + V_y(0)y + V_z(0)z \quad (4)$$

$$W(x, y, z) = W(0) + W_x(0)x + W_y(0)y + W_z(0)z \quad (5)$$

where $U(x, y, z)$, $V(x, y, z)$ and $W(x, y, z)$ are the three components of \vec{F} in x , y and z direction respectively (Eq. 2). The circular patch γ can be parameterized as follows:

$$\gamma : h(t) = (r\cos(t), r\sin(t), 0), \quad 0 \leq t \leq 2\pi \quad (6)$$

We now substitute $x = r\cos(t)$, $y = r\sin(t)$ and $z = 0$ in Eq. 3, 4 and 5 to obtain:

$$U(h(t)) = U(0) + U_x(0)r\cos(t) + U_y(0)r\sin(t) \quad (7)$$

$$V(h(t)) = V(0) + V_x(0)r\cos(t) + V_y(0)r\sin(t) \quad (8)$$

$$W(h(t)) = W(0) + W_x(0)r\cos(t) + W_y(0)r\sin(t) \quad (9)$$

The line integral of the vector field \vec{F} is next computed over this patch γ . The integrand in the line integral becomes:

$$\begin{aligned} F(h(t)) \cdot h'(t) &= r(V(0)\cos(t) - U(0)\sin(t)) + r^2(V_x(0)\cos^2(t) - U_y(0)\sin^2(t)) \\ &\quad + (V_y(0) - U_x(0)\sin(t)\cos(t)) + \dots \end{aligned} \quad (10)$$

In Eq. 10, the term (\dots) represents all the remaining parts from the higher order approximation. We next integrate Eq. 10 from 0 to 2π , and use the fact that:

$$\int_0^{2\pi} \sin(t)dt = \int_0^{2\pi} \cos(t)dt = \int_0^{2\pi} \sin(t)\cos(t)dt = 0 \quad (11)$$

$$\int_0^{2\pi} \sin^2(t)dt = \int_0^{2\pi} \cos^2(t)dt = \pi \quad (12)$$

We next obtain the circulation of the vector field \vec{F} around γ as follows:

$$\oint_{\gamma} \vec{F} \cdot d\gamma = \int_0^{2\pi} \vec{F}(h(t)) \cdot h'(t)dt = \pi r^2 (V_x(0) - U_y(0)) \quad (13)$$

where $h'(t)$ is the velocity vector field of the circular patch γ :

$$h'(t) = (-r\sin(t), r\cos(t), 0) \quad (14)$$

The 3D-Vor descriptor for the local surface γ is finally constructed by dividing the circulation of the vector field around this path by the area $A = \pi r^2$ of the circular patch, and taking the limit $r \rightarrow 0$:

$$3DVor = \lim_{r \rightarrow 0} \frac{1}{|A|} \oint_{\gamma} \vec{F} \cdot d\vec{r} \quad (15)$$

where $\oint_{\gamma} \vec{F} \cdot d\vec{r}$ in Eq. 15 is a line integral along the boundary of the curve γ and $|A|$ is the magnitude of the area. **Algorithm 1 provides the pseudo-code for 3D-Vor computation.** Fig. 3 illustrates the construction of the proposed feature descriptor. The circulation of the vector field over the local surface patch is shown in Fig. 3(c). Note that the vector field has a clockwise direction in the circular patch γ . The direction of the resulting vector normal \hat{n} (at P_k) is determined using the right hand rule. The latter indicates the vorticity of the vector field, as shown in Fig. 3(d) with different colors. As the circulation of the vector field is in the clockwise direction, \hat{n} is pointing downwards. The different values are concatenated into a vector to derive the 3D-Vor descriptor for the local surface patch, which has a fixed size for all the objects, shown in Fig. 3(e). As

Algorithm 1: Pseudo-code for 3D-Vor Computation.

Input: Detected feature points $P_k, k = 1, \dots, K$ on range image

Local surface patch L_k extracted around each P_k at radius r

Output: 3D-Vor *descriptor*

```
1 initialization;
2 for each  $L_k, k = 1, \dots, K$  do
3   // Computation of vector field
4    $F\{k\} \leftarrow \bar{U}\hat{x} + \bar{V}\hat{y} + \bar{W}\hat{z}$ 
5   // Compute circulation of the vector field
6    $\varepsilon\{k\} \leftarrow \oint_{\gamma} F\{k\} \cdot d\gamma$ . (See Eq. 13)
7   // Computation of vorticity
8    $Vor\{k\} \leftarrow \lim_{r \rightarrow 0} \frac{1}{|A|} \varepsilon\{k\}$ 
9   // Construct 3D-Vor (vector form)
10   $3D-Vor\{k\} \leftarrow Vor\{k\}$ 
11 end
```

Eq. 15 shows, 3D-Vor does not depend on the location of the coordinate axis (x, y, z) , in the Cartesian coordinates. Therefore, the resulting local surface feature is invariant to rotation and translation. Moreover, the reoriented vector field with respect to its unique local coordinate system (Section 3.1) also ensures the rotation invariance of the local features. Thus, if a patch is rotated by some angle, the local coordinate system along with the vector field also rotates by the same angle.

3.3. 3D-Vor Selection Parameter

As Eq. 15 shows, the 3D-Vor feature descriptor has only one critical parameter; the support radius r (or size of the local surface patch). The support radius r determines the portion of the underlying 3D surface that is captured by the 3D-Vor descriptor. The value of r can be chosen depending on how local the feature should be. A tradeoff is required between the feature’s descriptiveness and its computation time. **That is, a large support radius enables the 3D-Vor descriptor to capture more geometrical information of the object.** On the other hand, a large support radius significantly increases the computation time and the described area. We evaluated the performance of 3D-Vor descriptor with respect to varying support radius. The results are reported in Table 2.

The results show that the feature matching performance of the 3D-Vor descriptor improves as the support radius is increased from $5mr$ (mr = mesh resolution) to $25mr$.

Table 2: Support radii ‘ r ’ for 3D-Vor descriptor and associated computation time. We chose $r=10mr$ (where mr = mesh resolution) in this paper as a tradeoff between effectiveness and computational efficiency.

Support Radii r	Feature Matching Accuracy (%)	Computation Time (sec)
$5\ mr$	60	10.2
$10\ mr$	95	75.6
$15\ mr$	97	105.11
$20\ mr$	98	200.3
$25\ mr$	99	296.6
$30\ mr$	97	380.2

Specifically, there is a significant improvement of the matching performance as r increases from $5mr$ to $10mr$, this is because of the fact that a radius of $5mr$ is too small to capture significant distinctive information of the underlying 3D surface. The 3D-Vor descriptor achieved good results with a support radius of $10mr$, achieving 95% feature matching accuracy within a reasonable computation time. The performance of the 3D-Vor descriptor improves slightly as r is increased to $25mr$, at the expense of a significant increase in computation time. The performance of 3D-Vor slightly deteriorates when the support radius is set to $30mr$ and the computation time reaches 380.2s. We set the support radius to $10mr$ in this paper to maintain a fair tradeoff between the descriptor’s effectiveness and the computational complexity.

Note that, several adaptive-scale keypoint detection methods have been proposed to determine the support radius based on the inherent scale of a feature point. However, we chose to simply adopt a fixed support radius because our focus relates to the design of a novel feature descriptor for range image registration rather than feature point detection.

4. Automatic Pairwise Correspondence and Registration

In this section, we describe 3D-Vor based feature matching and automatic pairwise range image registration.

4.1. Feature Correspondence

Given a pair of 2.5D/partial views (V_1, V_2) of an object, we respectively extract a set of 3D-Vor descriptors from both range images. The 3D-Vor local descriptors extracted from a pair of views are then matched to find correspondences between the 3D views. Matching 3D-Vor descriptors are then used for the automatic coarse registration of the views by aligning the local coordinate systems used to derive these descriptors. Since a 3D-Vor is a 3D descriptor of a local surface patch of an object, a pair of matching 3D-Vors reveals that the surface patches represented by these 3D-Vors are similar and should correspond to the same surface patch of the 3D object. We use the correlation coefficient to match a pair of 3D-Vor descriptors:

$$C_c = \frac{n_v \sum_{i=1}^{n_v} u_i v_i - \sum_{i=1}^{n_v} u_i \sum_{i=1}^{n_v} v_i}{\sqrt{n_v \sum_{i=1}^{n_v} u_i^2 - (\sum_{i=1}^{n_v} u_i)^2} \sqrt{n_v \sum_{i=1}^{n_v} v_i^2 - (\sum_{i=1}^{n_v} v_i)^2}} \quad (16)$$

where, u_i and v_i ($i = 1, \dots, n_v$) are the elements of 3D-Vor from V_1 and V_2 respectively, in their region of overlap. n_v is the number of pairs of 3D-Vor descriptors used for feature matching.

To establish correspondence between view 1 (V_1) and view 2 (V_2), the overlap ratio R_o of the 3D-Vor descriptor of V_1 and V_2 is calculated first using Eq. 17.

$$R_o = \frac{m_t}{m_1 + m_2 - m_t} \quad (17)$$

In Eq.17, m_t is the number of non-zero elements of 3D-Vor from V_1 that have a corresponding non-zero element at the same index position in the 3D-Vor of V_2 . m_1 and m_2 are the total numbers of non-zero elements in the 3D-Vor descriptors of V_1 and V_2 , respectively. If R_o is greater than a pre-specified threshold t_r (equal to 0.4 in our experiments), the algorithm proceeds to calculate the correlation coefficient C_c (Eq. 16) of the two 3D-Vors in their region of overlap. Otherwise, the next 3D-Vor descriptor from V_1 is considered for matching. If C_c is greater than a pre-specified threshold t_c (which is set to 0.5), the algorithm proceeds to the next step of the local validation.

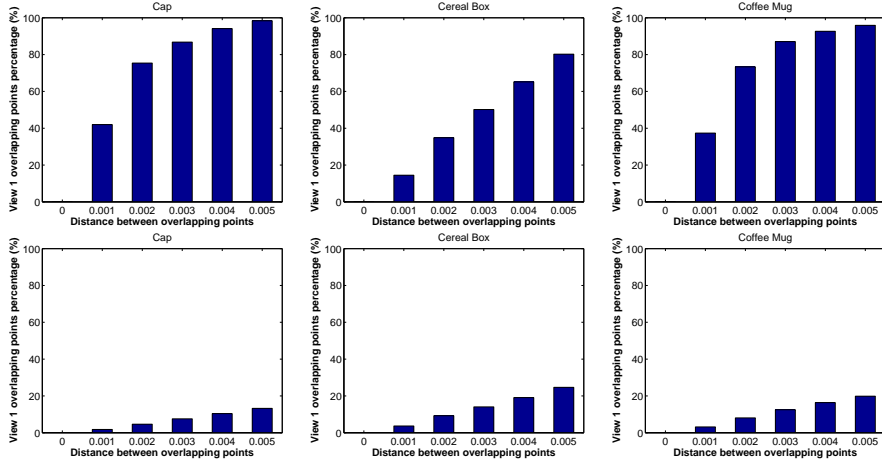


Figure 4: Outcome of the local validation step for the range images of the cap, the cereal box and the coffee mug (See Fig. 6). **First Row:** Accurate feature correspondences achieved in these cases. Most of the overlapping points are within the threshold t_d . **Second Row:** These results indicate a failure of the local validation due to inaccurate correspondences. In this case, a 3D-Vor pair was discarded and a new pair was tested.

4.2. Local Validation

During the local validation, all the points of V_2 are transformed to the coordinates of V_1 . This transformation is calculated by transforming the corresponding 3D local coordinate system of V_2 to the local coordinate system of V_1 using Eqs. 18 and 19.

$$R = L_1^T L_2 \quad (18)$$

$$t = p_1 - p_2 R \quad (19)$$

In Eq. 18, $L_k (k = 1, 2)$ is a 3×3 matrix of the x, y, z coordinate vectors of the local coordinate system. In Eq. 19, p_1 and p_2 are the positions of the corresponding points from V_1 and V_2 respectively. R and t are the rotation matrix and translation vectors, respectively. These transformations align V_2 with V_1 .

In the next step, the distance of every 3D point in V_1 is calculated to the nearest

neighbor 3D point in V_2 (which is already aligned with V_1 using Eq. 18 and 19):

$$d(q, p) = \sqrt{\sum_{i=1}^n (q_i - p_i)^2} \quad (20)$$

where p_i and q_i represent the 3D points of V_1 and V_2 respectively. n is the total number of points in V_1 . If the maximum distance (here in mm) between the two overlapping views is less than or equal to a specified threshold t_d and the number of corresponding points between the two views is more than or equal to n_k ($n_k = n_o/n_t$, where n_o is the number of overlapping points and n_t is the total number of 3D points), the transformation between V_1 and V_2 is accepted and V_2 is registered to V_1 . Otherwise the next pair of 3D-Vor descriptors is selected for matching. In our experiments, we set $t_d=0.003$ based on empirical tests. Fig. 4 shows the outcome of the local validation step for the range images of the cap, the cereal box and the coffee mug. Fig. 4 reports the percentage of 3D points on V_1 which successfully found their nearest neighbor points on V_2 for a specific distance threshold ($t_d=0.003$). Note that the local validation is done, once the transformation between V_1 and V_2 has been estimated and the two views have been aligned. Fig. 4 (first row) shows the result of a local validation that leads to accurate registration of the 3D views. For $t_d=0.003$, almost 70% (on average) of the 3D points on V_1 found their nearest neighbor points in V_2 . If the local validation steps described above fail, the next pair of the 3D-Vor descriptors is selected for matching and the whole process is repeated. Fig. 4 (second row) depicts the outcome of such cases, when a pair of 3D-Vor descriptors could not achieve accurate matching/transformation. In these cases, the 3D points on V_1 could not find a significant amount of nearest neighboring points in V_2 . These pairs of 3D-Vor descriptors were discarded and the next one were selected for matching. Note that in our approach, automatic coarse registration is performed on the basis of a single pair of matching 3D-Vor descriptors. An alternate possibility is to match a predetermined number of 3D-Vors and calculate the rigid trans-

Algorithm 2: Pseudo-code for Automatic Correspondence and Registration.

Input: Given 2.5D views V_1, V_2 of an object, and their respective 3D – Vor descriptors (in vector form)

Output: Transformation, Correspondences

```
1 initialization;
2 for each pair of overlapping views  $V_1$  and  $V_2$  do
3   for each 3D-Vor of  $V_1$  and  $V_2$  do
4     if  $R_0 \geq t_r$  AND  $C_c \geq t_c$  then
5       Transform  $V_2$  to  $V_1$ ;
6       if  $d(q, p) \leq t_d$  AND  $n_k \geq n_c$  then
7         RETURN Transformation and Correspondence
8       end
9     end
10  end
11 end
```

formation supported by the maximum number of matching 3D-Vors using RANSAC. This approach could be more robust but it will be computationally more expensive as there are six degrees of freedom when registering two range images. Moreover, the coarse registration resulting from a single pair of matching 3D-Vor descriptors is quite accurate (Fig. 5). It can thus serve as a reliable starting point for a subsequent refinement with a fine registration algorithm (e.g., ICP [28]). Algorithm 2 provides the pseudo-code of the proposed automatic correspondence and registration algorithm.

5. Experimental Results

We evaluated our 3D-Vor descriptor on a large-scale low resolution multi-view object dataset collected using Kinect sensor (Washington RGB-D (Kinect) object dataset) [29] to test the accuracy of 3D-Vor and its robustness to variations to pointcloud resolution. The dataset includes 300 daily use objects in 51 categories. Each category contains more than three objects and up to a maximum of twelve objects per category. The evaluation was done on all the 51 object categories and the achieved registration results are reported in the following sections. A performance comparison was also performed against the renowned spin image [30] and the recently proposed 3D SURF [5] and SISI/LD-SIFT descriptors [4].

5.1. Qualitative Analysis

We applied our 3D-Vor based pairwise range image registration technique to range images from the low resolution Washington RGB-D object dataset [29]. We used 600 **ordered** views of the objects in these experiments (i.e. 300 objects \times 2 views). **Note that the ordered set of views is the only input to our proposed automatic algorithm. The latter does not make any assumption about the viewing angles, the overlapping regions or the shape of the object, nor does the algorithm require any other information.** Figure 5 shows the feature correspondence and automatic pairwise coarse registration of range images on the basis of a single pair of matching 3D-Vors. It is clear from Fig. 5(a)-(d) that most of the feature correspondences are correct. Also, note that our coarse registration results in Fig. 5(e)-(h) are quite accurate even though no registration refinement had been performed at that stage. These results show the high descriptiveness of our proposed local surface descriptor.

Fig. 6 reports the qualitative results achieved by 3D-Vor based pairwise registration for 20 different objects taken from the Washington RGB-D (Kinect) object dataset [29]. Despite the fact that these range images of objects exhibit varying symmetries, features, curvatures, planar regions, the achieved registration accuracy is quite evident from Fig. 6.

5.2. Quantitative Analysis

In addition to the qualitative analysis, we also performed a quantitative analysis through extensive testing of our automatic pairwise correspondence and registration technique on the low resolution Washington RGB-D object dataset [29]. The proposed technique was tested according to the following criteria: (1) Accuracy (2) Robustness to resolution (3) Robustness to noise (4) Efficiency with respect to computation and memory and (5) Required overlap. Details are given below with respect to each criteria.

5.2.1. Accuracy

In these experiments, the accuracy of the proposed technique was evaluated for all the 51 object categories. The ground truth transformations calculated using our previous work [12] on the same dataset were used for evaluation purposes. The object categories contain range images of different types of objects with varying surface characteristics. Prominent examples include bowls, apples, coffee mug and cap images which have high curvatures. Food box and note book have planar regions, while hand towel, noodle packets and the sponge have varying symmetries from different views. The transformations (R_i and t_i) were evaluated for each

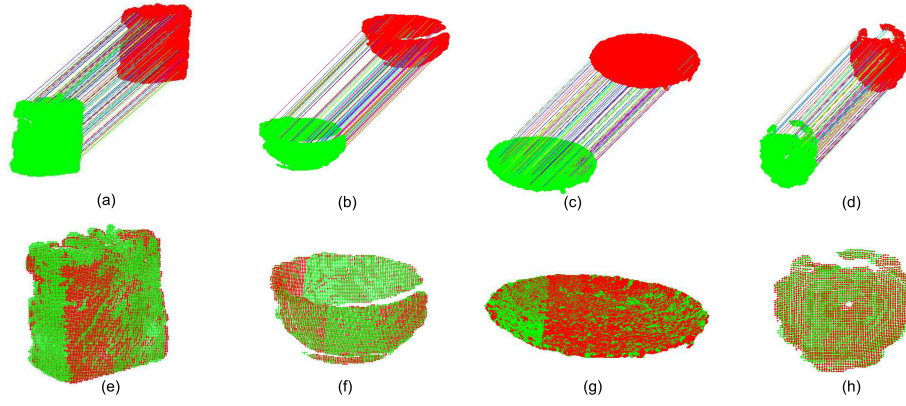


Figure 5: Feature correspondence and pairwise range image registration results. (a)-(d) Feature matching results between two range images of the Food box, the Plate, the Bowl and an Apple. (e)-(h) Automatic pairwise (coarse) registration results. Note that our coarse registration results are quite accurate even though calculated from a single pair of matching 3D-Vors in each case. No registration refinement (e.g., with ICP algorithm) was performed (This figure is best viewed in color).

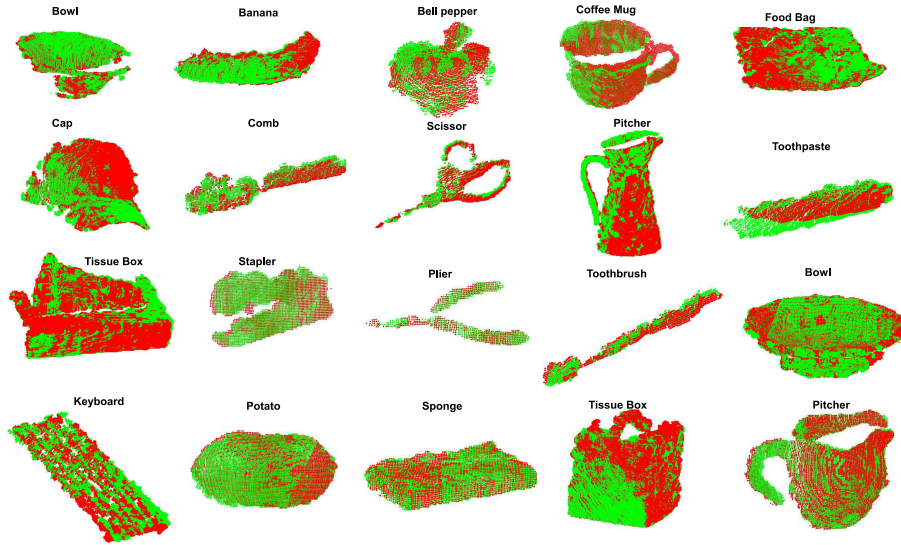


Figure 6: Pairwise range image registration results using the proposed 3D-Vor descriptor for 20 different objects. (Figure best seen in color).

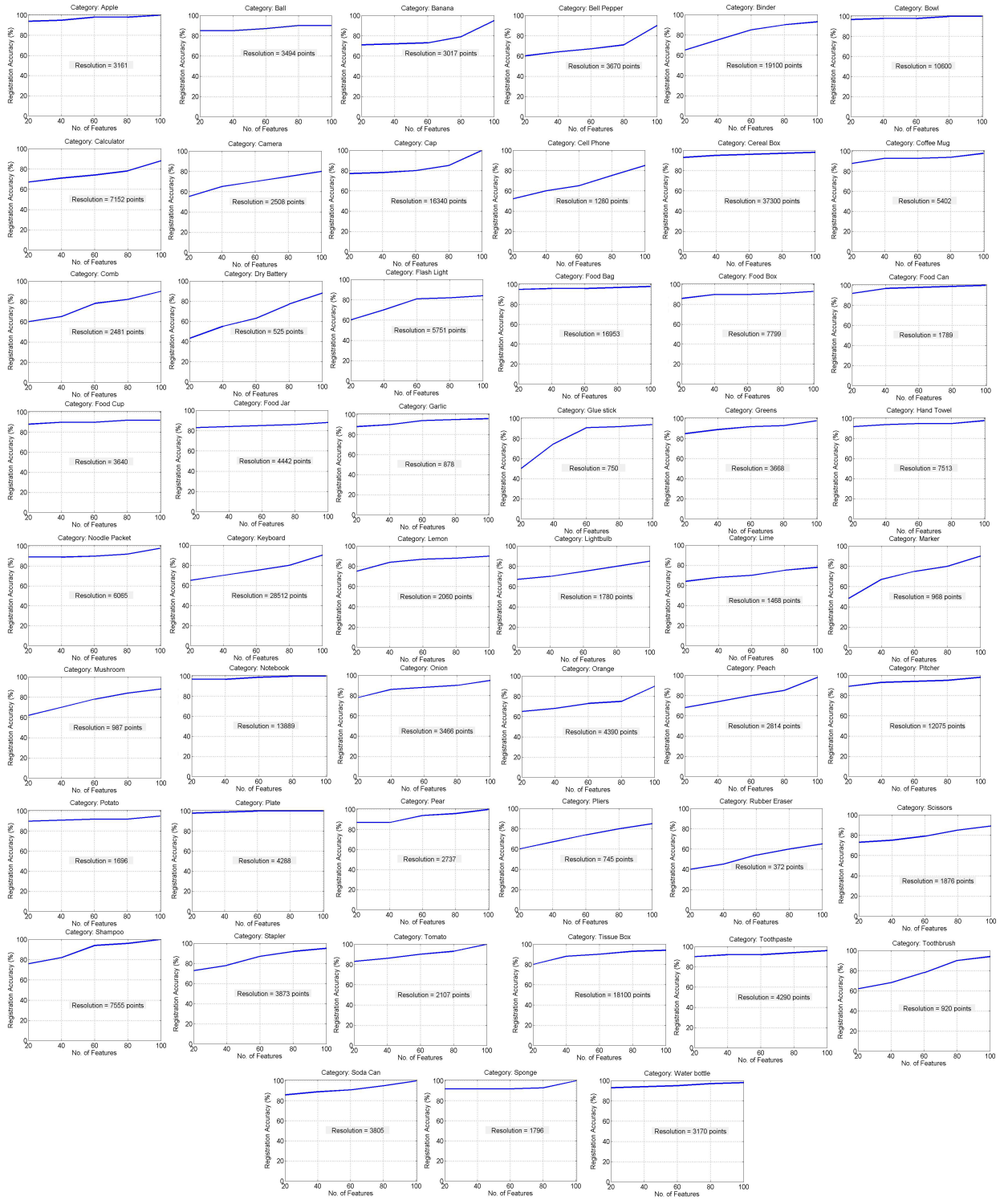


Figure 7: Registration accuracy achieved by 3D₁₀ descriptor for 51 different object categories. Each category has more than three objects with different pointcloud resolutions.

object category using our 3D-Vor based automatic registration algorithm. These transformations were then compared with the ground truth transformations (R_{iGT} and t_{iGT}). The error of the two rotation matrices was calculated as [1]:

$$R_{id} = R_i R_{iGT}^{-1} \quad (21)$$

$$\theta_{ie} = \cos^{-1} \left(\frac{\text{trace}(R_{id} - 1)}{2} \right) \frac{180}{\pi} \quad (22)$$

In Eq. 21, R_{id} is the rotation matrix denoting the difference between R_i and R_{iGT} . R_{id} is equal to the identity matrix in the case of zero error [1]. θ_{ie} in Eq. 22 represents the amount of rotation error present in R_i , and is derived from Rodrigue’s formula [1]. Similarly, the translation error t_{ie} of each view is calculated as follows:

$$t_{id} = \frac{\|t_i - t_{iGT}\|}{mr} \quad (23)$$

In Eq.23, mr is the mesh resolution of the 3D pointcloud. Fig. 7 reports the registration accuracy achieved by the 3D-Vor descriptor for all the 51 categories of objects [29]. Different views of the objects were registered using the pairwise range image correspondence/registration technique discussed in Section 4. Note that our 3D-Vor descriptor achieves more than 95% registration accuracy for most of the object categories, with only 100 3D-Vor descriptors per view. These results clearly demonstrate the superior performance of the proposed descriptor.

5.2.2. Robustness to variations in Resolution

To evaluate the robustness of 3D-Vor descriptor to varying mesh resolutions, we resampled the partial (2.5D) views of the objects to $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$ and $\frac{1}{16}$ of their original mesh resolution. The registration accuracy achieved by the proposed descriptor under different levels of mesh decimation is presented in Fig. 8. The accuracy decreases as the mesh resolution is varied from mr to $\frac{1}{16}mr$. Note that the overall accuracy, however,

remains above 87% even at a mesh resolution of $\frac{1}{16}mr$.

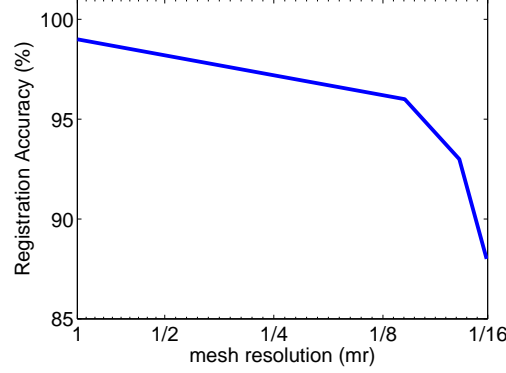


Figure 8: Robustness of the proposed descriptor to variations in mesh resolution.

5.2.3. Robustness to Noise

We used the range images of the cap, the coffee mug, the tissue box and the food box for these experiments. Gaussian noise with standard deviation $\sigma = mr, 2mr, 4mr, 6mr, 8mr$ and $10mr$ (where mr is the mesh resolution of the range image) was added to these range images. Next our 3D-Vor based algorithm was used to automatically register overlapping views. Fig. 9(a) shows two different views of the cap. Noise with $\sigma = 10mr = 2.8$ cm has been added to these views in Fig. 9(b). Registration results are shown in Fig. 9(c). We can see that the registration is correct even though most of the features on each view of the cap, the coffee mug, the tissue box and the food box were distorted due to the addition of the noise. The robustness of our technique to noise can be attributed to the following reasons: the locally tangent circulation of the vector field results in a high signal-to-noise ration. Moreover, the integration operation carried out in the computation of vorticity has a smoothing effect, providing increased robustness to noise. In addition to this, our algorithm uses the correlation coefficient for matching the 3D-Vor descriptors. The correlation coefficient being a statistical measure [1], performs better in the presence of noise compared to other matching techniques

Table 3: Time taken to register partial views of different resolutions

Object	Pointcloud Resolution	Registration Time (sec) (Feature Correspondence and Alignment)
Rubber eraser	372	1.04
Pliers	745	3.71
Potato	1696	3.85
Shampoo bottle	7555	4
Tissue Box	18100	4.1

(e.g., linear matching).

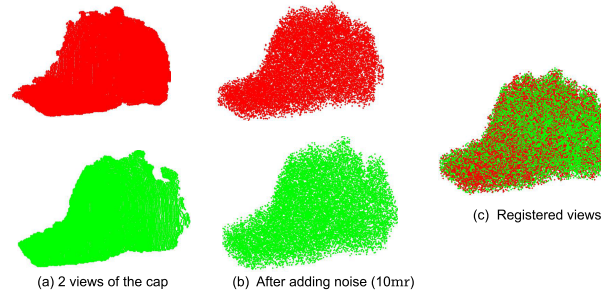


Figure 9: Robustness to noise results. (a) Two 3D views of the cap (noise free). (b) After adding Gaussian noise with $\sigma = 10mr$, most of the features of the cap are distorted. (c) The noisy views can still be correctly registered. (Figure is best viewed in color).

5.2.4. Efficiency with Respect to Computation and Memory

For a given support radius r , the computational efficiency is achieved by only matching a small number of 3D-Vor descriptors. Our experiments show that most of the time a correct pair of matching 3D-Vors is found when the first few 3D-Vor descriptors of V_1 are matched with the 3D-Vors of V_2 . Once a correct match (i.e., the first successful match) passing the local validation is found, the transformation between the views is accepted and our algorithm stops searching for any further matches, hence saving computational time. Table 3 reports the registration time taken to register partial views of different resolutions. These experiments were conducted on a computer with an Intel Corei3 CPU and 4GB RAM. Our code was simply implemented in MATLAB without any parallel computing technique. The details about the computation time of 3D-Vor and its comparison with state-of-the-art is provided in Section 6 (Fig. 10(right)).

Our proposed technique is also efficient in terms of memory utilization because a limited and constant number of 3D-Vor descriptors (approx. 100) are required to represent each view. Furthermore, the small size of the 3D-Vor descriptor ($10mr$ in these experiments) further cuts down the memory utilization.

5.2.5. Required Overlap

In this experiment, we tested the performance of our proposed algorithm for varying amounts of overlap between the range images to be registered. These experiments were performed on the range images of the cap, the coffee mug, the cereal box and the stapler [29]. We define the amount of overlap between two range images V_1 and V_2 as follows:

$$Overlap = \frac{No. \text{ of corresponding 3D points of } V_1 \text{ and } V_2}{\min(No. \text{ of 3D points in } V_1, No. \text{ of 3D points in } V_2)} \quad (24)$$

For each of the four objects, the overlap was calculated between all possible $N(N - 1)/2$ pairs of views (N is the total number of views per object). The overlap was calculated after the views were registered using a transformation calculated from our previous work [12]. Next, we used our automatic algorithm for pairwise registration of each of the $N(N - 1)/2$ view pairs and categorized the results as correct or incorrect. We noted that generally an overlap of 60% or more ensures a correct match.

6. Comparison with State-of-the-art

We compared our proposed descriptor with state-of-the-art techniques. Fig. 10(left) reports quantitative results related to the accuracy of the proposed automatic registration approach based on 3D-Vor descriptor against the popular spin image [30] (which is one of the most cited methods [11]) and the recently proposed 3D SURF [5] and SISI/LD-SIFT descriptors [4]. The number of descriptors per view were varied from 20 to 100 and the accuracy of the pairwise range image registration was recorded. The proposed local surface descriptor achieves an overall accuracy of 90% under varying pointcloud sizes, while spin image, 3D SURF and SISI/LD-SIFT descriptors achieved

an accuracy of 50%, 69.2% and 52% respectively. These results clearly demonstrate the robustness of our proposed descriptor to low resolution data. We also tested the performance of 3D-Vor with the feature point detector in [27]. In this case, 3D-Vor based registration achieved an accuracy of 87.6%. The reason behind this slight reduction in accuracy is the low repeatability of feature point detector [27] on low resolution data, compared to [4]. The proposed 3D-Vor descriptor captures the distinctive characteristics of the underlying 3D local surface by computing vorticity per unit area at every point of the field. The robustness to mesh resolution is achieved by using all the points of the 3D local surface for feature representation. The achieved accuracy can be increased further by extending the size of the support radius r , as explained in Section 3.3. However, this will also increase the computation time of the algorithm.

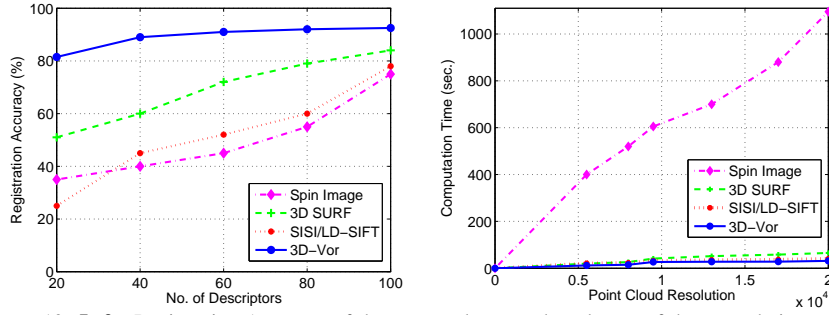


Figure 10: **Left.** Registration Accuracy of the proposed approach and state-of-the-art techniques. **Right.** Comparison of the computation time taken by 3D-Vor, spin images, 3D-SURF and SISI/LD-SIFT.

The reason behind the degraded accuracy of the spin images with low resolution data is that the descriptiveness of the spin images is dependent on its bin size [30]. Bin size is an important parameter for spin image generation. The bin size is set as a multiple of the mesh resolution [30]. As the latter is related to the density of points in the mesh and to the size of the shape features in the object, the descriptiveness of the spin image is adversely affected by a low resolution dataset. The same explanation applies to the low performance of SISI descriptor [4], which is the extension of spin images. LD-SIFT descriptor [4] also could not achieve a satisfactory performance for the low

resolution dataset. It is because of the fact that LD-SIFT is computed by representing the neighborhood of the feature point as a depth map. The robustness of the descriptor is highly dependent on the local scale parameter, that defines the neighborhood of the feature point [4]. Due to the low resolution data, the SISI/LD-SIFT descriptor of [4] cannot capture the significant surface information. The 3D SURF descriptor also suffered from a relatively low descriptiveness for a low resolution data. It is because of the fact that 3D SURF is based on the concept of Haar wavelet responses, stored at each grid cell [5]. The technical disadvantage of the Haar wavelet is that it is not continuous and therefore not differentiable [31]. This problem becomes more prominent with low resolution data which contains holes and missing depth information (i.e. discontinuity). Due to this reason, the 3D SURF achieved a lower registration accuracy.

Fig. 10 (right) compares the computational efficiency of the proposed technique with the state-of-the-art. The average time taken by our technique is 21sec, compared to 700sec, 43.6sec and 31.8sec for spin images, 3D-SURF and SISI/LD-SIFT, respectively. These results clearly demonstrate the computational efficiency of our proposed technique. All the algorithms were implemented in MATLAB. The computational efficiency of our proposed technique is expected to improve manifolds once it is implemented in C++.

7. Multiview Range Image Registration (3D Modeling)

To demonstrate the effectiveness of 3D-Vor for 3D modeling, we next performed automatic multiview registration of range images $\{V_1, V_2, \dots, V_N\}$ of a 3D object. We used a tree based approach to perform multiview range image registration. We selected the range image V_i as the root node of the tree. We then used the pairwise range image registration technique described above to match the 3D-Vor descriptors and to register V_i with V_j . Once range image V_j is accurately registered with V_i , the range image V_j is added to the tree as a new node. The rigid transformation between V_i and

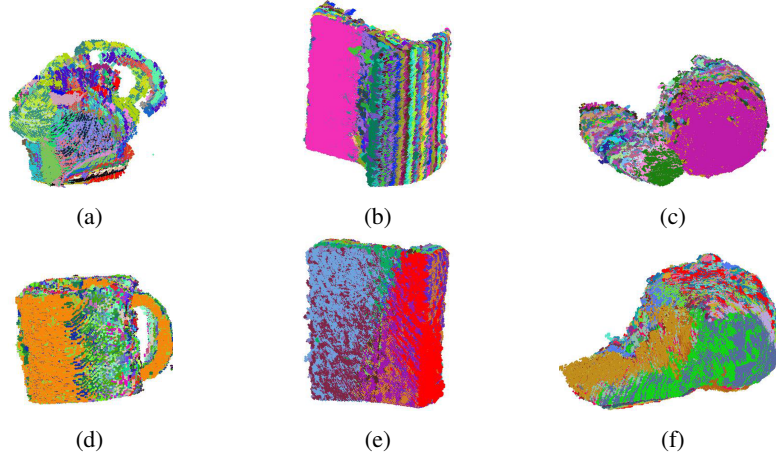


Figure 11: Multiview range image registration (3D Modeling) results. (a)-(c) Initial set of 40 range images of the Coffee Mug, Cereal Box, Cap (unregistered). (d)-(f) Coarse registration result of our proposed local surface feature based method. (Figure is best viewed in color).

V_j is represented by an arc between the two nodes. Once the tree is fully constructed by registering all the range images using the aforementioned process, the transformation between any two nodes is then made available. Based on these estimated transformations, all range images are transformed to the same coordinate system of the range image at the root node. These multiview range images can therefore be aligned without any user/manual intervention.

We applied our local surface descriptor based multiview range image registration approach to 40 range images of the objects [29]. Fig. 11(a)-(c) shows the initial set of range images for the coffee mug, the cereal box and the cap. While Fig. 11(d)-(f) gives the coarse registration results for multiple views of these objects. It is clear that our local feature based technique achieves a highly accurate alignment with low resolution 3D data.

8. Conclusion

In this paper, we presented 3D-Vor, a novel rotation invariant local surface descriptor which exploits the vorticity of the vector field for surface representation. The vorticity is described by a resulting vector at each point of the local surface. The attributes of this vector (magnitude and direction) are concatenated to derive the proposed 3D-Vor descriptor. Such a feature representation is shown to provide superior surface

descriptiveness, which in turn results into an accurate pairwise registration. We also presented a fully automatic correspondence and registration algorithm by efficiently matching 3D-Vors of overlapping views. The proposed algorithm does not require any prior knowledge of the viewing angles or the regions of overlap of the views. We evaluated the performance of the proposed algorithm and the local surface descriptor on the challenging low resolution Washington RGB-D (Kinect) object dataset. Reported experimental results show that the proposed local surface descriptor is robust and more accurate compared to state-of-the-art approaches. It achieves 90% registration accuracy compared to 50%, 69.2% and 52% achieved by spin image, 3D SURF and SISI/LD-SIFT descriptors, respectively.

Acknowledgments

This research is supported by RTS scholarship from the University of Western Australia (UWA) and Australian Research Council (ARC) grant DP110102166.

References

- [1] A. S. Mian, M. Bennamoun, R. A. Owens, A novel representation and feature matching algorithm for automatic pairwise registration of range images, *International Journal of Computer Vision* 66 (2006) 19–40.
- [2] S. A. A. Shah, K. M. Yahya, G. Mubashar, A. Bais, Quantification and visualization of MRI cartilage of the knee: A simplified approach, in: *IEEE 6th International Conference on Emerging Technologies (ICET)*, 2010, pp. 175–180.
- [3] S. A. A. Shah, M. Bennamoun, F. Boussaid, A. El-Sallam, Automatic object detection using objectness measure, in: *ICCSA*, 2013, pp. 1–6.
- [4] Y. Keller, T. Darom, Scale invariant features for 3D mesh models, *IEEE Transaction on Image Processing* 21 (2012) 2758–2769.
- [5] J. Knopp, M. Prasad, G. Willems, R. Timofte, L. Gool, Hough transform and 3D-SURF for robust three dimensional classification, in: *Computer Vision ECCV 2010*, volume 6316 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2010, pp. 589–602.
- [6] B. Kim, H. Yoo, K. Sohn, Exact order based feature descriptor for illumination robust image matching, *Pattern Recognition* 46 (2013) 3268 – 3278.
- [7] C. Li, J. Li, D. Gao, B. Fu, Rapid-transform based rotation invariant descriptor for texture classification under non-ideal conditions, *Pattern Recognition* 47 (2014) 313 – 325.
- [8] M. Hasegawa, S. Tabbone, Amplitude-only log radon transform for geometric invariant shape descriptor, *Pattern Recognition* 47 (2014) 643 – 658.
- [9] C. Geng, X. Jiang, Face recognition based on the multi-scale local image structures, *Pattern Recognition* 44 (2011) 2565 – 2575.

- [10] X. Song, D. Muselet, A. Trémeau, Affine transforms between image space and color space for invariant local descriptors, *Pattern Recognition* 46 (2013) 2376–2389.
- [11] Y. Guo, F. Sohel, M. Bennamoun, M. Lu, J. Wan, Rotational projection statistics for 3D local surface description and object recognition, *International Journal of Computer Vision* 105 (2013) 63–86.
- [12] S. A. A. Shah, M. Bennamoun, F. Boussaid, A. A. El-Sallam, 3D-DIV: A novel local surface descriptor for feature matching and pairwise range image registration, in: *IEEE ICIP*, pp. 2934–2938.
- [13] S. A. A. Shah, M. Bennamoun, F. Boussaid, A. El-Sallam, A novel local surface description for automatic 3d object recognition in low resolution cluttered scenes, in: *ICCVW*, 2013, pp. 638–643.
- [14] S. A. A. Shah, M. Bennamoun, F. Boussaid, Performance evaluation of 3d local surface descriptors for low and high resolution range image registration, in: *Digital Image Computing: Techniques and Applications (DICTA)*, 2014 International Conference on, pp. 1–7.
- [15] A. Mademlis, P. Daras, D. Tzovaras, M. G. Strintzis, 3d object retrieval using the 3d shape impact descriptor, *Pattern Recognition* 42 (2009) 2447–2459.
- [16] Y. Gao, Q. Dai, N.-Y. Zhang, 3d model comparison using spatial structure circular descriptor, *Pattern Recognition* 43 (2010) 1142–1151.
- [17] M. Kolomenkin, I. Shimshoni, A. Tal, Prominent field for shape processing and analysis of archaeological artifacts, *International Journal of Computer Vision* 94 (2011) 89–100.
- [18] M. S. Longuet-Higgins, Vorticity and curvature at a free surface, *Journal of Fluid Mechanics* 356 (1998) 149–153.
- [19] I. Stamos, M. Leordeanu, Automated feature-based range registration of urban scenes of large scale, in: *CVPR*, 2003. Proceedings. 2003 IEEE, volume 2, IEEE, pp. II–555.
- [20] J. Salvi, C. Matabosch, D. Fofi, J. Forest, A review of recent range image registration methods with accuracy evaluation, *Image and Vision Computing* 25 (2007) 578 – 596.
- [21] C.-S. Chen, Y.-P. Hung, J.-B. Cheng, RANSAC-based DARCES: A new approach to fast automatic registration of partially overlapping range images, *PAMI*, *IEEE Transactions on* 21 (1999) 1229–1234.
- [22] J.-P. Tarel, H. Civi, D. B. Cooper, Pose estimation of free-form 3D objects without point matching using algebraic surface models, in: *Proceedings of IEEE Workshop on Model-Based 3D Image Analysis* 1998, pp. 13–21.

- [23] K. Brunnstrom, A. J. Stoddart, Genetic algorithms for free-form surface matching, in: *Pattern Recognition, 1996.*, Proceedings of the 13th International Conference on, volume 4, IEEE, pp. 689–693.
- [24] J. V. Wyngaerd, L. Van Gool, R. Kock, M. Proesmans, Invariant-based registration of surface patches, in: *IEEE International Conference on Computer Vision, 1999.*, volume 1, IEEE, pp. 301–306.
- [25] K. Higuchi, M. Hebert, K. Ikeuchi, Building 3D models from unregistered range images, *Graphical models and image processing* 57 (1995) 315–333.
- [26] G. Roth, Registering two overlapping range images, In *IEEE International Conference on 3-D Digital Imaging and Modelling* (1999) 191–200.
- [27] A. Mian, M. Bennamoun, R. Owens, On the repeatability and quality of keypoints for local feature-based 3D object retrieval from cluttered scenes, *International Journal of Computer Vision* 89 (2010) 348–361.
- [28] P. J. Neugebauer, Reconstruction of real-world objects via simultaneous registration and robust combination of multiple range images, *International Journal of Shape Modeling* 3 (1997) 71–90.
- [29] K. Lai, L. Bo, X. Ren, D. Fox, A large-scale hierarchical multi-view RGB-D object dataset, in: *IEEE ICRA, 2011*, pp. 1817–1824.
- [30] A. Johnson, M. Hebert, Using spin images for efficient object recognition in cluttered 3D scenes, *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 21 (1999) 433–449.
- [31] B. Vidakovic, *Statistical modeling by wavelets*, volume 503, John Wiley & Sons, 2009.