Consistent Pose Normalization of Non-Rigid Shapes using One-Class Support Vector Machines

Panagiotis Papadakis, Fiora Pirri

HAL Id: hal-00758993

https://inria.hal.science/hal-00758993

Submitted on 29 Nov 2012

Consistent Pose Normalization of Non-Rigid Shapes using One-Class Support Vector Machines

Panagiotis Papadakis, Fiora Pirri
Auto Agent Laboratory for Cognitive Robotics
Department of Informatics and Systems, University of Rome 'La Sapienza'
papadakis@dis.uniroma1.it, pirri@dis.uniroma1.it

Abstract

The estimation of 3D surface correspondence constitutes a fundamental problem in shape matching and analysis applications. In the presence of non-rigid shape deformations, the ambiguity of surface correspondence increases together with the complexity of registration algorithms. In this paper, we alleviate this problem by using One-Class Support Vector Machines (OCSVM) in order to normalize the pose of 3D objects. We show how OCSVM are employed in order to increase the consistency of translation and scale normalization under articulations, extrusions or the presence of outliers. To estimate the relative translation and scale of an object, we use the 3D distribution of points that is modelled by employing OCSVM to estimate the decision surface corresponding to the surface points of the object. To evaluate the performance, we use a dataset of 3D objects where we introduce various extrusions, articulations or outliers and demonstrate the increased robustness of the proposed methodology.

1. Introduction

The increasing availability of 3D content together with the advent of affordable 3D acquisition technology has stimulated research in methods that facilitate the subsequent processing of 3D objects. Within the plurality of applications where 3D objects are used, one of the most commonly encountered problems is pose normalization which decomposes into the normalization of the translation, scale and rotation. Pose normalization is most often performed in applications such as content-based retrieval, thumbnail generation, visualization and modeling where 3D objects need to be positioned in a canonical frame. Pairwise surface registration provides the optimal solution for the shape correspondence problem which is drawing an increasing research interest and has recently been introduced in SHREC [1]. Pose normalization is an alternative approach that pro-

vides a single, global solution for each 3D object instance which results in significant efficiency gains at the cost of suboptimal surface correspondence.

In this paper, we focus on the translation and scale normalization components of the 3D pose normalization problem that have been relatively undervalued in comparison to rotation normalization. However, all three components are equally important to obtain consistent shape correspondence. We demonstrate that the state-of-the-art approaches for translation and scale normalization become insufficient as class-variation increases due to non-rigid shape deformations such as articulation, extrusion or the presence of isolated-outlying parts. To alleviate this problem, we propose the usage of a *novelty detection* algorithm, namely, the One-Class Support Vector Machines (OCSVM) [12, 14] as a more robust approach. Using OCSVM we estimate the support of the high-dimensional distribution corresponding to a set of points sampled from the surface of a 3D object and derive the underlying distribution corresponding to the object as the 3D volume that is constrained within the boundaries of the OCSVM decision surface in 3D space. This distribution is then used to derive the relative translation and scale of the object in order to perform a more consistent normalization. This is achieved by lowering the sensitivity of normalization to relatively trivial parts of the object through outlier identification and employing a volume-based approach that is more descriptive compared to solely considering the surface of the object.

The remainder of the paper is organized as follows: In Section 2, we discuss related work in the area of translation and scale normalization and outline the limitations of the state-of-the-art approaches in order to motivate the proposed methodology. In Section 3, we formally describe the context in which we employ OCSVM to estimate the distribution of a 3D object that is used to derive the corresponding normalizing translation and scale. And finally, in Section 4, we provide a quantitative evaluation that demonstrates the superiority of the proposed method.

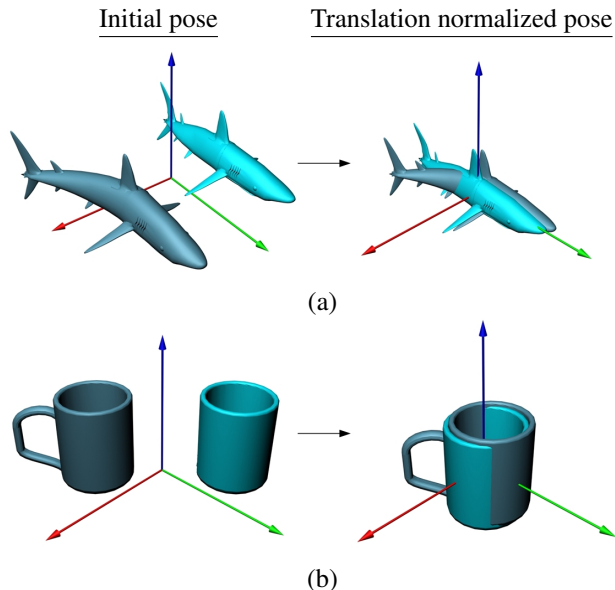Initial pose   Translation normalized pose

(a)

(b)

Figure 1. Translation normalization using the centroid of the surface of the object; (a) Example in articulated and non-articulated object and (b) in an object with and without extruding parts.

2. Related work

3D pose normalization accounts for the normalization of the translation, scale and rotation of 3D objects. This step is required in applications where 3D objects need to be positioned in a canonical coordinate frame as, in general, these three characteristics are arbitrarily set. For example, it is necessary in content-based retrieval, where shape matching is based on the establishment of correspondences between the surfaces of objects. Among the three normalization problems, it is generally admitted that rotation normalization is the most challenging problem with a significant amount of related work [19, 8, 10], where the spatial, orientation or symmetry distribution of the surface has been used to determine the rotation of the frame of the object. However, since rotation normalization is performed on the basis of an appropriately chosen center of an object, translation normalization determines the effectiveness of pose normalization to a greater extent. Therefore, we focus on the translation normalization of 3D objects as well as scale normalization that has been relatively undervalued although it is a difficult problem in a number of cases.

Translation Normalization This problem is most frequently addressed by computing the center of mass of a 3D object and by translating it to make its center coincide with the coordinates origin. For 3D geometrical objects consisting of polygons, the center of the object is computed as a weighted average of the total set of the polygon vertices [20], or centroids of the polygons [19], i.e. as the centroid of its surface. For the majority of 3D objects, this method
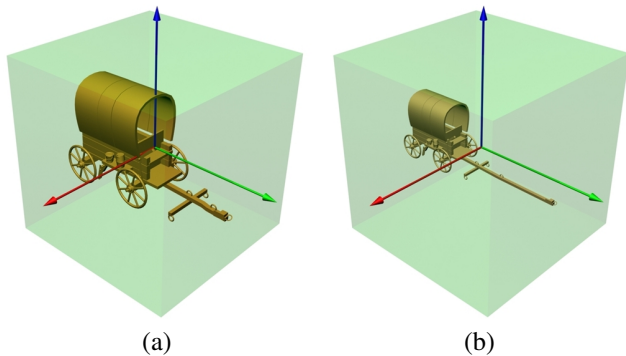

(a)   (b)

Figure 2. Scale normalization to the unit cube. The scale normalized version of the 3D object shown in (a) is not consistent in terms of shape correspondence with the scale normalized version of its deformed version with elongated carriage axis in (b).

is effective and objects are well normalized with respect to translation. However, there are cases where this method is not appropriate such as articulated 3D objects or objects with isolated-outlying or extruding parts. Such cases are demonstrated in Fig. 1 (a) and (b) respectively. Intuitively, we would agree that the centers of the two objects in either case should coincide (in a semantic context). However, by setting the centroid of the surface of an object as its center, we obtain a result that does not agree with our perception. As it is evident from the examples, a small change in the shape of 3D objects can have significant impact on the resulting translation normalization which may in turn negatively affect scale and rotation normalization. Clearly, we expect that the perturbation of the centroid of the surface of an object becomes more pronounced as the amount of articulation or extrusion increases.

In the work of Podolak et al [10], the authors propose the center of symmetry computed by the Planar Reflective Symmetry Transform as an alternative approach to compute the center of an object. They show that this method is more robust in consistently determining the center of 3D scans of objects, however the performance of the method depends on the symmetry properties of the objects that become less descriptive in the presence of non-rigid transformations.

The smallest enclosing ball of an object [5] has also been considered as a method to determine its center as well as its scale [7]. However, this approach is directly dependent to the outlying parts of an object which makes it the most unstable method as will be demonstrated at Section 4.

Scale Normalization This problem is traditionally addressed by setting the scale of a 3D object to fit within a bounding volume of fixed size (unit cube or unit sphere), or to a size proportional to the average distance of the surface of an object from its centroid [8, 19].

The performance of the first method is problematic in the presence of outlying parts that will cause an undesired

shrinking of the entire object in order to make it fit within the bounding volume. A characteristic example is shown in Fig. 2 where the objects are scaled so as to fit within the unit cube. The object in Fig. 2 (b) is the same in every aspect as the object in Fig. 2 (a) not considering the elongated carriage axis. However, the scale normalized versions of the two objects are significantly different.

Although scaling to a size proportional to the average distance of the surface of an object from its centroid alleviates this problem, it is more a heuristic than a sound methodology. In fact, the consistency of the normalization can only be evaluated by visually examining each 3D object individually. Moreover, the computation of the average distance is still dependent on the entire surface which implies that the effect of trivial outlying parts is not trivial if they are distant from the main cluster of points (e.g. Fig. 2 (b)).

3. Pose Normalization using One-class Support Vector Machines

One-class support vector machines (OCSVM) is a *novelty detection* algorithm [12] that estimates the support of a high-dimensional distribution characterised by a set of points labeled as belonging to the distribution. It has been employed for applications that range from classification [3, 6, 9, 17] to medical imaging [21], outlier detection [22, 16] and surface reconstruction [13], but this is the first time that it is employed to address the translation and scale normalization of 3D objects.

3.1. Distribution estimation of a 3D object

We use OCSVM to estimate a 3D distribution corresponding to the surface of a 3D object that enables a more consistent computation of its relative translation and scale. In detail, we consider the surface of a 3D object as a collection of 3D points that are sampled from the corresponding 3D distribution and we use OCSVM [2] to compute the decision surface that separates this distribution from the residual Euclidean space. The distribution that corresponds to the object is the volume that is constrained within the boundaries of the decision surface.

Let $X = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m\}$ denote the set of points that are densely sampled from the surface of a 3D object, where m is the number of points, $\mathbf{x}_i \in T$ and $T \subseteq \mathbb{R}^3$. Let Φ be a feature mapping from space T to a dot product space H and $k(\mathbf{x}, \mathbf{y})$ a kernel function such that $\langle \Phi(\mathbf{x}), \Phi(\mathbf{x}')\rangle = k(\mathbf{x}, \mathbf{x}')$, where $\langle ., .\rangle$ denotes the dot product in H. We want to compute a hyper-sphere with center $\mathbf{c}$ and radius R that contains the maximum number of points $\mathbf{x}_i$ while having the minimum possible radius. This enables identifying parts that are comparatively small to the overall shape and distant from the main cluster of points that could cause a drifting in the computation of the relative translation and scale.

The computation of the hyper-sphere is formulated as an optimization problem described as follows:

$$\min_{\mathbf{c}\in H, R\in\mathbb{R}} R^2 + \frac{1}{vm}\sum_{i=1}^{m}\xi_i,$$
$$subject\ to\ \|\Phi(\mathbf{x}_i) - \mathbf{c}\|^2 \leq R^2 + \xi_i \qquad (1)$$

Here ξ_i denote the "slack" variables that permit the exclusion of outliers from the interior of the hypersphere, i.e. parts that are relatively trivial if they reside far from the primary distribution of points. Slack variables enable dealing with non-rigid shape deformations such as extrusions and articulations or isolated parts of the object that ideally should not affect the determination of the relative translation and scale. The parameter $v \in (0, 1]$ is used to control the amount of slack, in other words, the tolerance to outliers. By decreasing the value of v, we increase the tolerance of the algorithm to outliers, while increasing the value of v we achieve the opposite effect.

To solve the problem, we convert the objective function to its dual form and derive the Lagrangian, from which we get the dual problem:

$$\min_{\alpha} \sum_{i,j=1}^{m} \alpha_i\alpha_j k(\mathbf{x}_i, \mathbf{x}_j) - \sum_{i=1}^{m}\alpha_i k(\mathbf{x}_i, \mathbf{x}_i),$$
$$subject\ to\ 0 \leq \alpha_i \leq \frac{1}{vm}\ and\ \sum_{i=1}^{m}\alpha_i = 1 \qquad (2)$$

We solve eq. (2) by employing Sequential Minimal Optimization [11], from which we obtain the coefficients α_i and derive the center $\mathbf{c}$ of the hyper-sphere. The decision function that models the distribution of points is determined by $f_s(\mathbf{x}) = sgn(f(\mathbf{x}))$ where sgn is the sign operator, giving $+1$ if the input is positive, -1 if it is negative and 0 otherwise, and $f(\mathbf{x})$ is given by:

$$
\begin{aligned}
f(\mathbf{x}) &= R^2 - (\mathbf{c} - \Phi(\mathbf{x}))^2 \\
&= R^2 - \left[\sum_{i=1}^{m}\alpha_i\Phi(\mathbf{x}_i)\right]^2 + 2\sum_{i=1}^{m}\alpha_i\Phi(\mathbf{x}_i)\Phi(\mathbf{x}) - [\Phi(\mathbf{x})]^2 \\
&= R^2 - \sum_{i,j=1}^{m}\alpha_i\alpha_j k(\mathbf{x}_i, \mathbf{x}_j) + 2\sum_{i=1}^{m}\alpha_i k(\mathbf{x}_i, \mathbf{x}) - k(\mathbf{x}, \mathbf{x})
\end{aligned}
\qquad (3)
$$

If $f_s(\mathbf{x}) = +1$ then $\Phi(\mathbf{x})$ lies inside the hypershere and $\mathbf{x}$ is considered to belong to the estimated distribution, while if $f_s(\mathbf{x}) = -1$, $\Phi(\mathbf{x})$ lies outside the hypersphere and $\mathbf{x}$ is considered not to belong to it. If $f_s(\mathbf{x}) = 0$ then $\mathbf{x}$ lies on the decision surface, as in the case of support vectors.

In general, OCSVM can be employed using different kernels and settings of parameters that determine the final shape of the decision surface in the observation space.

In our work we use the Gaussian kernel $k_G(\mathbf{a}, \mathbf{b}) = e^{-\gamma||\mathbf{a}-\mathbf{b}||^2}$, where $\mathbf{a}, \mathbf{b} \in T$, since it has the property that any set of points sampled from the surface of a 3D object will always be separable in the projection (feature) space, i.e. there is always a solution. This is true since the dot product of any two mapped points is always positive, which implies that all mapped points lie on the same orphant of the projection space. Furthermore, all mapped points have unit length which ensures that they are separable from the origin in the projection space [13].

Using the gaussian kernel the OCSVM algorithm is parameterized by the γ variable of the kernel and the variable v that controls the slack. The γ variable controls the shape of the decision surface in 3D space in terms of its similarity with respect to the sphere and the underlying surface of the object. In particular, as $\gamma \to 0$ the decision surface approaches the shape of the sphere, while, as γ increases, the decision surface fits better to the surface of the object. This can be better understood by considering that γ is inversely proportional to the σ^2 of the gaussian kernel, thus, when $\sigma \to 0$ the uncertainty is decreased and the decision surface tends to fit the surface of the object. The v variable controls the amount of points from the surface of the object that will reside within the decision surface and it can be used to improve the robustness of the algorithm with respect to articulated, extruding or outlying parts. As $v \to 0$ outliers are minimized while the opposite is true as $v \to 1$. In Fig. 3 we demonstrate the effect of different γ and v settings in estimating the distribution of 3D objects.
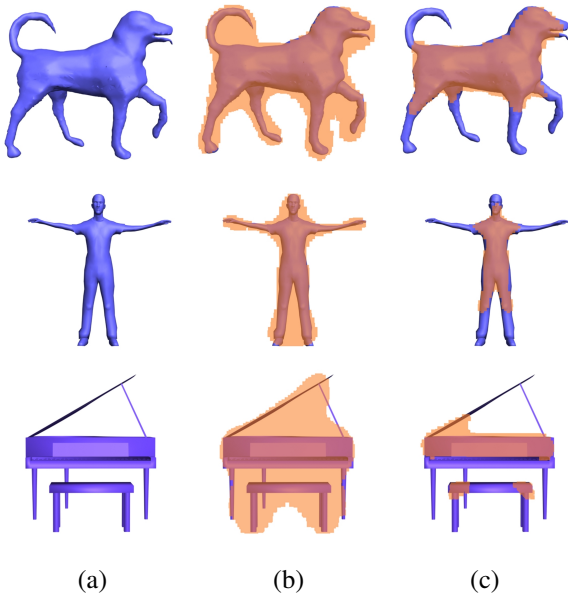

(a) (b) (c)

Figure 3. (a) Example 3D objects and their estimated distribution (depicted in orange) using OCSVM for (b) $\gamma = 15$, $v = 0.02$ and (c) $\gamma = 45$, $v = 0.8$.

3.2. Translation and Scale Normalization

OCSVM methodology applied to translation normalization let us overcome the limitations of the state-of-the-art approaches that cannot face increasing intra-class variation of 3D objects. The insufficiency is due to the assumption that the center of a 3D object is the center of mass of the surface of the object. This approach is not robust when non-rigid shape deformations are applied to 3D objects, such as articulations, extrusions or the presence of outlying parts (see Fig. 1) where the center of mass of the surface can be considerably perturbed.

The solution that we propose provides an alternative method for computing the center of an object that is less sensitive to the aforementioned scenarios. More specifically, an object center is taken to be the centroid $\mathbf{c}_m$ of the distribution estimated according to the OCSVM method, and whose volume V is within the boundaries of the decision function $f_s(\mathbf{x})$ in the observation space. The centroid of the distribution is computed as:

$$\mathbf{c}_m = \frac{1}{V} \int \int \int_{f_s(\mathbf{p}) \geq 0} \mathbf{p} \, dx dy dz \qquad (4)$$

where $\mathbf{p} = (x, y, z) \in T$ and $V = \int \int \int_{f_s(\mathbf{p}) \geq 0} dx dy dz$.

The motivation behind the proposed approach for computing the object center is two-fold. First, by using OCSVM we are able to decrease the sensitivity to extruding-outlying parts by appropriately setting the slack, i.e. the v variable. And second, we adopt a volume-based approach for computing the centroid which is more stable than only considering the surface area of the object. This is achieved by considering not only the decision surface, i.e. where $f_s(\mathbf{x}) = 0$ but also the space where $f_s(\mathbf{x}) = +1$. Furthermore, this enables the computation of a volumetric center even for objects that are not watertight.

After translation normalization, we can use the computed distribution that corresponds to the 3D object to perform scale normalization. To normalize the scale we estimate the average distance d_m of the distribution of the object from its center and scale the object so that this distance is unit. The average distance d_m is computed as:

$$d_m = \frac{1}{V} \int \int \int_{f_s(\mathbf{p}) \geq 0} \| \mathbf{p} - \mathbf{c}_m \| \, dx dy dz \qquad (5)$$

where $\| . \|$ denotes the L_2 norm.

The resulting scale normalization will be more consistent as by employing OCSVM with a preset tolerance to outliers we succeed in lowering the sensitivity of the normalization to trivial parts of the objects. Furthermore, we directly benefit from the more consistent determination of the center of an object using OCSVM since the average distance is computed with respect to this point.

Figure 4. The dataset of 3D objects used for evaluating the performance of translation and scale normalization. Each row corresponds to a class of 3D objects created by generating articulations, extrusions or adding-removing outlying parts.

4. Results

In this Section we evaluate the performance of the proposed methodology in terms of the consistency of pose normalization in the presence of outliers or when non-rigid transformations such as articulations and extrusions are applied to a variety of 3D objects.

4.1. Evaluation dataset

To evaluate the performance of pose normalization, we have generated a set of categorized 3D objects (Fig. 4) that contains 120 generic 3D objects in total, evenly distributed in 12 classes, namely, *human*, *dolphin*, *chair*, *tree*, *pot*, *desktop computer*, *hand*, *dog*, *guitar*, *piano*, *spider* and *horse*. Each class of the set was created using a reference 3D object. To build the dataset, we introduce various non-rigid deformations to the reference object of each class and generate various articulations, extrusions or add-remove outlying parts. By applying such modifications to the reference object, we create new instances of the same object which altogether constitute a particular class. Since the geometry of the generated objects is different than the

geometry of the respective reference object, we expect that the geometric center of the objects should also be altered. However, we do not apply any *similarity transformation* to the generated 3D objects, therefore, the objects of the same class are pose normalized with respect to their rigid part. This is required in order to evaluate pose normalization and to our knowledge there is no publicly available dataset that satisfies this requirement, therefore standard databases such as [15], [18] or [4] cannot be used for this evaluation.

4.2. Evaluation of translation and scale normalization

By comparing the standard deviation for the centroid of the surface of an object and the centroid of the distribution of the object as computed by OCSVM, in each class of the dataset, we can evaluate the performance of OCSVM under different settings of the γ parameter of the gaussian kernel and the slack v. With respect to the γ parameter, we have evaluated the performance for $\gamma \in (0, 50]$. The value of the upper limit was determined considering the effect of overfitting the estimated distribution to the points of the 3D object.
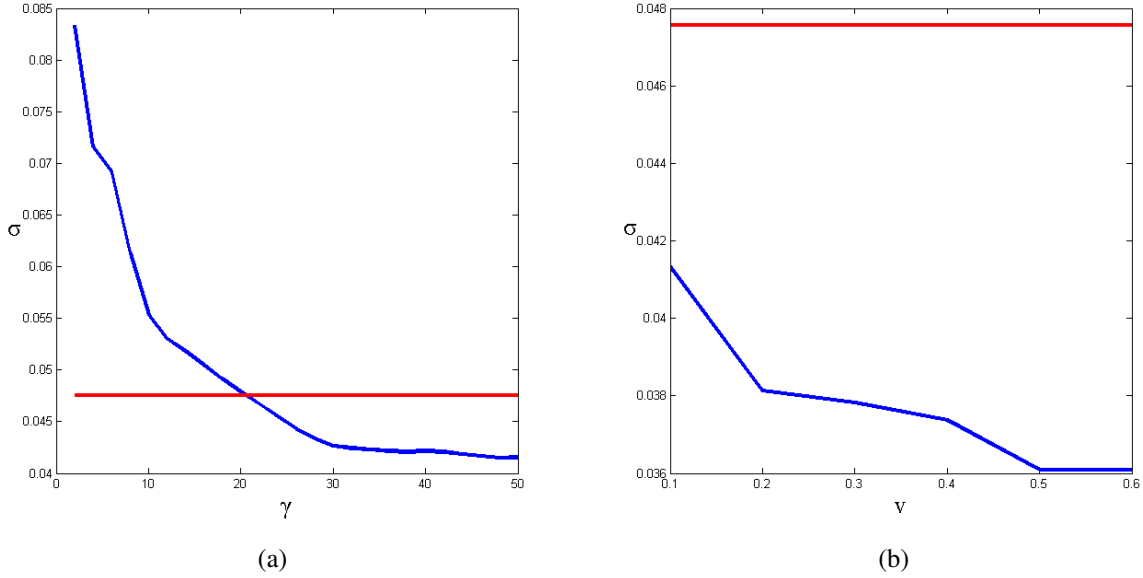
(a)   (b)

Figure 5. Average standard deviation of the centroid of an object computed using OCSVM (blue curve) for (a) $\gamma \in (0, 50]$ and $v = 0$ and (b) $v \in [0.1, 0.6]$ and $\gamma = 47$, compared to the standard deviation of the centroid of the complete surface of an object (red curve).

In Fig. 5 (a) we demonstrate how the standard deviation of the center of the distribution computed using OCSVM varies in relation to the γ parameter and compare it against the standard deviation of the centroid of the surface of an object. The performance is averaged over all classes. This evaluation demonstrates the increased consistency of translation normalization achieved by OCSVM solely due to the determination of the center by a volume-based approach rather than a surface-based approach, thus the slack is set to $v = 0$ and the complete 3D object surface is taken into account. The diagram shows that the perturbation of the center of the OCSVM distribution initially decreases rapidly as γ increases and finally converges to a global minimum. The point where the curve corresponding to OCSVM is minimized in Fig. 5 (a), i.e. the point where the average perturbation of the center is minimum, is for $\gamma = 47$. We interpret this point as the best trade-off between precisely characterizing the surface of an object through the OCSVM distribution and overfitting the distribution to the corresponding surface. Indirectly, Fig. 5 (a) also provides an evaluation for the consistency of translation normalization by translating an object to the center of its minimum enclosing sphere (i.e. when $\gamma \to 0$). It is evident that this method exhibits the worst performance compared to the surface centroid-based approach and the proposed approach using OCSVM.

After fixing $\gamma = 47$ we evaluate the performance of OCSVM translation normalization under different settings of the slack and in particular for $v \in [0.1, 0.6]$, as shown in Fig. 5 (b). The value of the upper limit implies that this is the boundary that we are using to detect outlying parts that should not be considered in the computation of the center.

This evaluation demonstrates the increased consistency of translation normalization that is achieved by OCSVM due to the determination of the center of an object by discarding outlying parts rather than considering the complete surface of an object. The point where the curve corresponding to OCSVM is minimized in Fig. 5 (b), i.e. the point where the average perturbation of the estimated center is minimum, is for $v = 0.5$. We interpret this point as the best trade-off between discarding outliers of a surface that should not be considered in the computation of its center and at the same time not discarding the important parts.

In Fig. 6, we evaluate the performance within each class of the dataset. Within classes where objects have a well defined core part (human, dolphin, chair, vase, computer, hand, dog, guitar, piano, spider and horse) as well as relatively smaller articulated-outlying parts, we observe that it is meaningful to introduce the slack in the OCSVM distribution estimation and discard a certain proportion of the surface of an object as being an outlier in order to determine its center. An exception is the tree class where the determination of the core part is more ambiguous. Here we observe the inverse behaviour since OCSVM may erroneously discard important parts of an object that are not outliers. In this case, the perturbation of the center increases as meaningful parts are not taken into account, however, the performance gain that is attained due to the volume-based approach for computing the center mostly compensates for this insufficiency. Generally, the performance of the proposed approach is stable for 3D objects where a core part can be identified and the remaining parts of the objects can be discarded up to a predefined ratio.
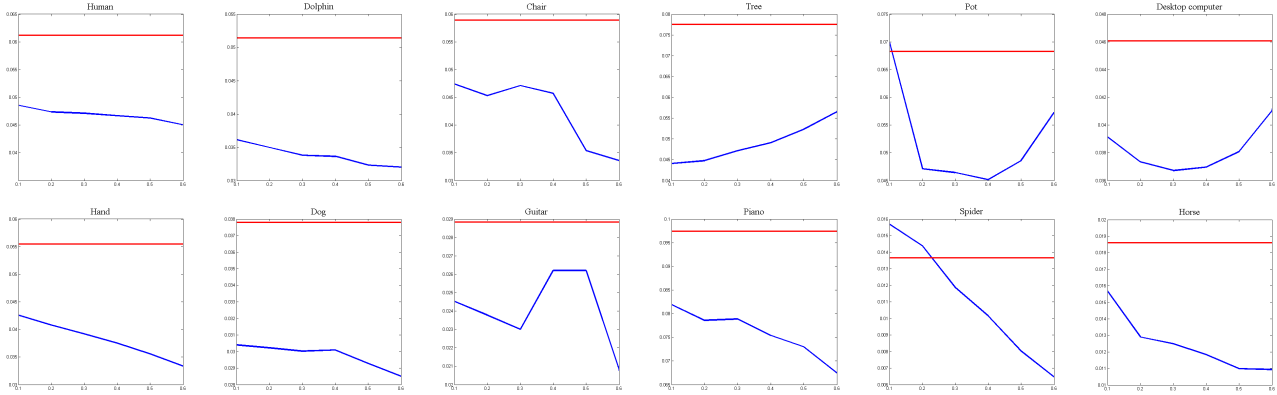
29

Figure 6. Average standard deviation of the centroid of an object computed using OCSVM (blue curve) for $v \in [0.1, 0.6]$ and fixed $\gamma = 47$, compared to the standard deviation of the centroid of the complete surface of a 3D object (red line).

By setting $\gamma = 47$ and $v = 0.5$ the average standard deviation of the OCSVM center of an object is equal to $\sigma_{tr,OCSVM} = 0.0361$ while the average standard deviation of the centroid of the surface of an object is $\sigma_{tr,centroid} = 0.0475$. In other words, we have lowered the average perturbation of the center for objects that belong to the same class by 24%. This performance gain does not only improve translation normalization, but also scale normalization which is based on measuring distances from the center of the object. Analogously, we expect that rotation normalization methods that involve computations with respect to the center of an object would benefit as well.

To compare the performance of OCSVM scale normalization against the method that uses the average distance of the surface from its centroid, we measure the standard deviation of the distance for each class and the overall performance is taken by averaging over all classes (Fig. 7).

The results demonstrate that the proposed method tends to be more robust than the standard approach since on aver-

age $\sigma_{scale,OCSVM} = 0.01425$ and $\sigma_{scale,stan} = 0.01699$. This means that we have achieved to lower the perturbation of the average distance computation by about 16.1%. The consistency of scale normalization when using OC-SVM is higher in 8 out of the 12 classes. The inferior performance in the other 4 classes can be partly attributed to the fact that for scale normalization using OCSVM we set the parameters v and γ to those values that gave the best results for translation normalization. Although this is not a strict a requirement, we adopt this approach in order to reduce the complexity of the total normalization process to a single run of the OCSVM algorithm. Alternatively, we could determine the values of v and γ based on the combined performance of translation and scale normalization.

In Fig. 8 we show the result of both translation and scale normalization for pairs of objects of the same class in order to provide a qualitative evaluation of the proposed approach, in terms of shape correspondence.

The presented results show that the problem of translation and scale normalization of 3D objects under non-rigid shape transformations such as articulations, extrusions or the presence of outliers can be effectively addressed using the proposed methodology that is based on One-Class Support Vector Machines. As part of our future work, we are considering the application of OCSVM to improve rotation normalization methods that are based on measuring 3D spatial features as we believe that measuring distances with respect to the more robustly defined center of 3D objects would increase the consistency of rotation normalization.
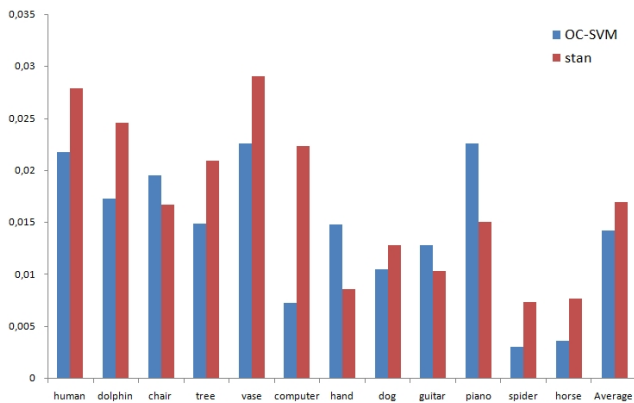

Figure 7. Evaluation of scale normalization by measuring the standard deviation (vertical axis) of the average distance from the centroid of the distribution using OCSVM and the average distance from the centroid of the surface of the objects (stan).

References

[1] A. M. Bronstein, M. M. Bronstein, U. Castellani, A. Dubrovina, L. J. Guibas, R. P. Horaud, R. Kimmel, D. Knossow, E. von Lavante, D. Mateus, M. Ovsjanikov, and A. Sharma. Shrec 2010: robust correspondence benchmark. In *Proc. EUROGRAPHICS Workshop on 3D Object Retrieval*, 2010.
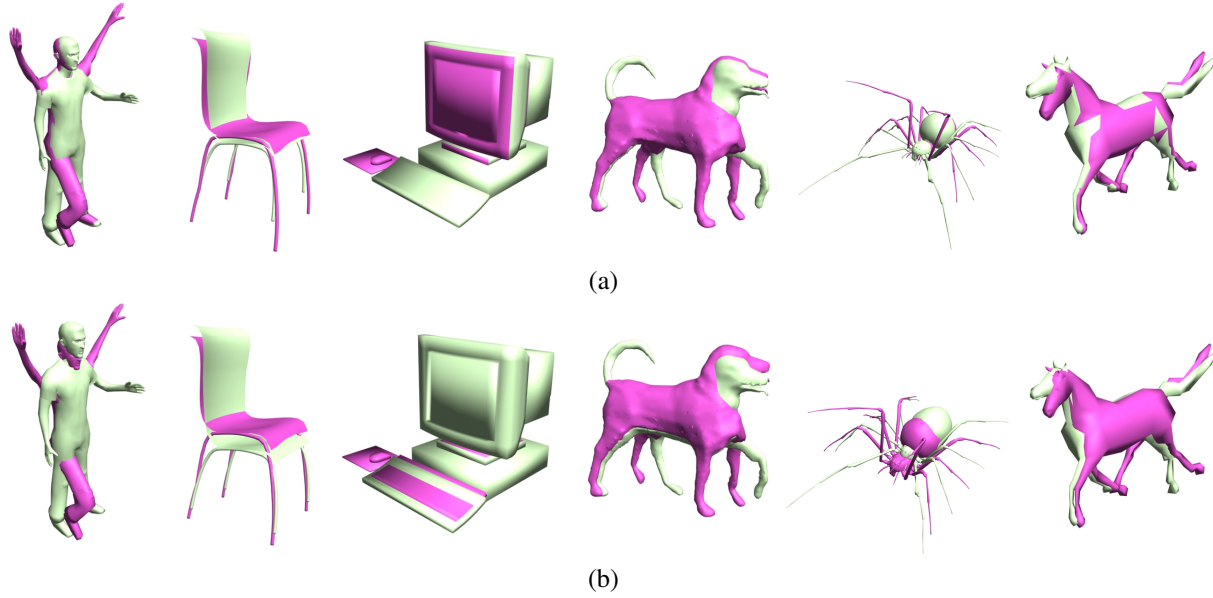
Figure 8. Translation and scale normalization using (a) OCSVM and (b) using the centroid and average distance of the surface.

[2] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[3] Y. Chen, X. S. Zhou, and T. Huand. One-class svm for learning in image retrieval. In *Proceedings of the Int. Conf. on Image Processing*, 2001.

[4] R. Fang, A. Godill, X. Li, and A. Wagan. A new shape benchmark for 3D object retrieval. In *International Symposium on Advances in Visual Computing*, 2008.

[5] K. Fischer and B. Gartner. The smallest enclosing ball of balls: Combinatorial structure and algorithms. *International Journal of Computational Geometry and Applications*, 14(4-5):341–378, 2004.

[6] M. Manevitz and Y. Malik. One-class svms for document classification. *Journal of Machine Learning Research*, 2:139–154, 2002.

[7] T. Napoleon and H. Sahbi. From 2d silhouettes to 3d object retrieval: Contributions and benchmarking. *EURASIP Journal on Image and Video Processing*, 2010:17 pages.

[8] P. Papadakis, I. Pratikakis, S. Perantonis, and T. Theoharis. Efficient 3D shape matching and retrieval using a concrete radialized spherical projection representation. *Pattern Recognition*, 40(9):2437–2452, 2007.

[9] P. Papadakis, I. Pratikakis, T. Trafalis, T. Theoharis, and S. Perantonis. Relevance feedback in content-based 3d object retrieval: A comparative study. *Computer-Aided Design and Applications*, 5(5):753–763, 2008.

[10] J. Podolak, P. Shilane, A. G. S. Rusinkiewicz, and T. Funkhouser. A planar-reflective symmetry transform for 3d shapes. *ACM Transactions on Graphics*, 25:549–559, 2006.

[11] F. Rong-En, C. Pai-Hsuen, and L. Chih-Jen. Working set selection using second order information for training support vector machines. *The Journal of Machine Learning Research*, 6:1889–1918, December 2005.

[12] B. Scholkop, J. Platt, J. Shawe-Taylor, A. Smola, and R. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13:1443–1471, 2001.

[13] B. Scholkopf, J. Giesen, and S. Spalinger. Kernel methods for implicit surface modeling. In *Advances in Neural Information Processing Systems*, pages 1193–1200, 2005.

[14] B. Scholkopf and A. Smola. *Learning with Kernels*. The MIT Press, 2002.

[15] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser. The princeton shape benchmark. In *Shape Modeling International*, pages 167–178, 2004.

[16] S. Taeshik and M. Jongsub. A hybrid machine learning approach to network anomaly detection. *Information Sciences*, 177(18):3799–3821, 2007.

[17] S. Tzikopoulos, M. Mavroforakis, H. Georgiou, N. Dimitropoulos, and S. Theodoridis. A fully automated scheme for mammographic segmentation and classification based on breast density and asymmetry. *Computer Methods and Programs in Biomedicine*, 102:47–63, 2011.

[18] R. Veltkamp and F. ter Haar. Shrec2007 3d shape retrieval contest. In *Technical Report, Department of Information and Computing Sciences, Utrecht University*, 2007.

[19] D. V. Vranic. *3D Model Retrieval*. PhD thesis, University of Leipzig, 2004.

[20] D. V. Vranic and D. Saupe. 3D model retrieval. In *Spring Conf. on Computer Graphics and its Applications*, 2000.

[21] L. Wang, G. Xu, L. Guo, X. Liu, and S. Yang. 3D reconstruction of head mri based on one class support vector machine with immune algorithm. In *Proceedings of the IEEE Int. Conf. on EMBS*, 2007.

[22] Y. Zhang, N. Meratnia, and P. Havinga. Adaptive and online one-class support vector machine-based outlier detection techniques for wireless sensor networks. In *Proceedings of the IEEE Int. Conf. on Advanced Information Networking and Applications Workshops*, pages 990–995, 2009.