

# Thesis Literature Review

Michał Staniaszek

February 25, 2015

## 1 Segmentation

Popularised by Comaniciu [11] for use in image segmentation, mean shift was first introduced by Fukunaga [13] in 1975, and rediscovered by Cheng [7] in 1995. The technique finds stationary points in a density estimate of the feature space, for example pixel RGB values, and uses those points to define regions in the space by allocating pixels to them. Pixels which follow the gradient of the density to the same stationary point are part of the same segment. An example can be seen in Figure 1.

## 2 Interest Points

Mikolajczyk and Schmid [20] introduce an interest point detector which is an improvement on SIFT [18] and the Laplacian of Gaussians [17] in the sense that it is able to deal with affine transformations. They do not, however, introduce a new type of descriptor to go with the point selection.

## 3 2D Descriptors

Different considerations must be made for 2D images because of affine transforms, rotation and so on

Van Gool [34] gives a description of how to use moment invariants to recognise planar patterns like labels and signs under affine deformations. The moments describe things like the size of the shape and its centre of mass, or statistics like the mean and variance of pixel intensities in the shape. These moments can be combined in such a way that they are invariant to deformations, which is useful to have in a descriptor.

## 4 3D Descriptors

The Ensemble of Shape Functions (ESF) descriptor introduced in [35] by Wohlkinger and Vincze combines the Shape Distribution approach introduced by [24] along with some extensions proposed in [16]. It also makes use of their voxel-based distance measure from [36]. Pairs or triples of points are sampled from segmented partial clouds of objects, and histograms are created by extracting information such as distance, angle, ratios, and whether points are inside or outside (or a mix) of the model. See Figure 2.

The Point Feature Histogram (PFH) was introduced by Rusu et al. in [29]. It creates descriptors based on the angles between a point on a surface and  $k$  points close to it. The Fast Point Feature Histogram (FPFH) improved the speed of computation, and allowed the use of the descriptor in real time [27]. The Viewpoint Feature Histogram (VFH) extended the FPFH by adding viewpoint information to the histogram by computing statistics of surface normals relative to the viewpoint [28]. It also improved the speed of the FPFH. The clustered version (CVFH) further improved the viewpoint technique by mitigating the effect of missing parts and extending it to be able to estimate rotations of objects [1].

Multi-scale descriptors are useful as they can be used to characterise regions of varying size. Cipriano et al. introduce such a descriptor for use on meshes [10]. It captures the statistics of the shape of the neighbourhood of a vertex by fitting a quadratic surface to it. Vertices in the region are weighted based on distance from an initial vertex, and a plane is constructed using a weighted average of the face normals. The parameters of the quadratic are then used to find its principle curvatures, which make up the descriptor.

Work in protein-protein docking also uses 3D descriptors to help with simulations of an otherwise lengthy and complex process. The Surface Histogram is introduced by Gu et al. [15], and uses the local geometry around two points with specific normals on the surface of a protein. A coordinate system is

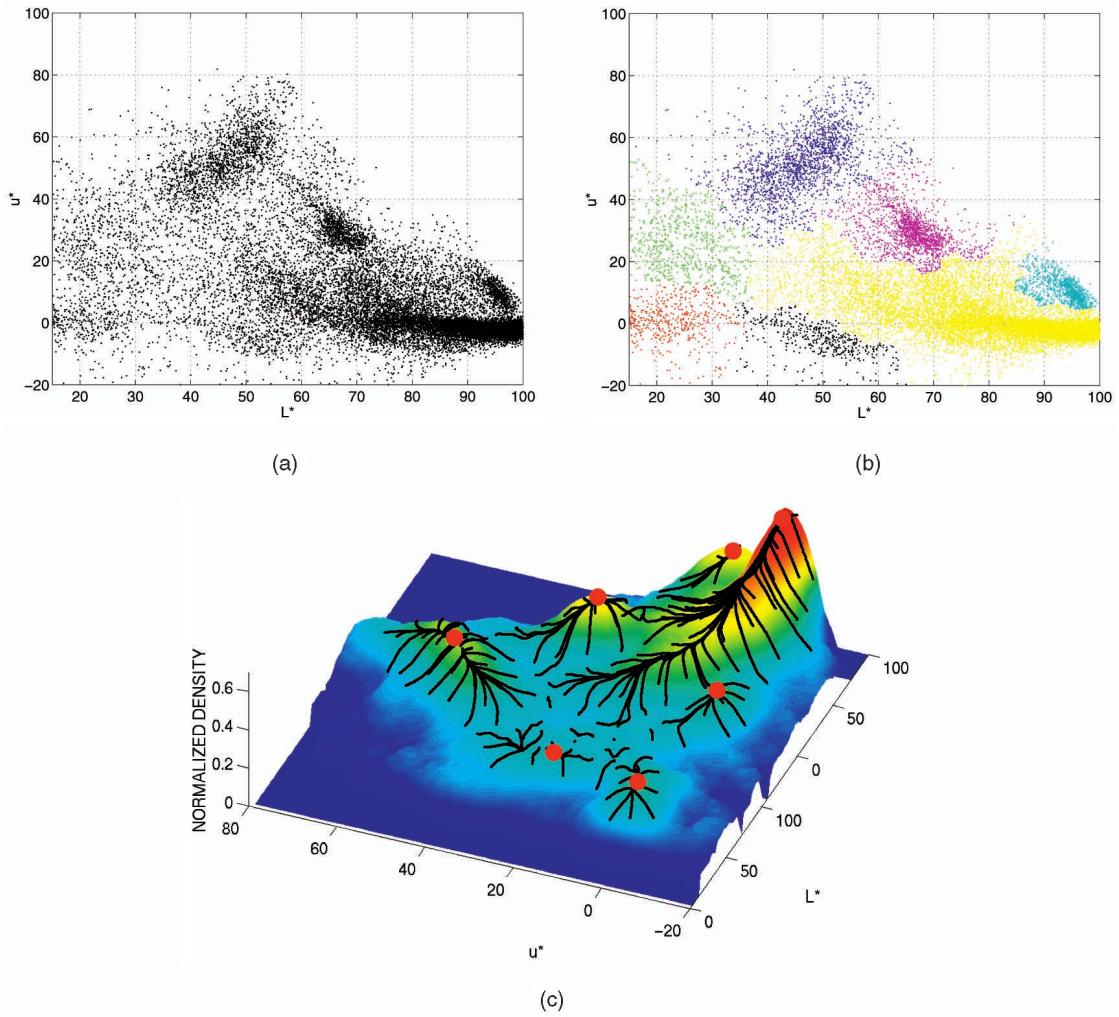


Figure 1: Visualisation of mean shift [11]. a) First two components of image pixels in LUV space. b) Decomposition found by running mean shift. c) Trajectories of mean shift over the density estimate.

defined by the two points and the line between them, and a rectangular voxel grid is defined around the points. The grid is then marked in locations where the surface crosses the grid, and a 2D image is constructed by squashing the data onto one of the axes. The descriptor is designed to immediately give a potential pose for the docking.

Another example of a shape descriptor from biology is the Context Shape [30]. A sphere is centred on a point, and rays are projected from this point to points evenly distributed on the surface of the sphere. Each of the rays is divided into segments, with a binary value associated with each segment depending on whether the segment is inside or outside the protein. To compare the descriptor, a rotation is applied to match the rays, and a volume of overlap is computed based on matching bits in the rays.

The splash descriptor was introduced by Stein et al. [33]. A point on the surface with a given surface normal (the reference normal) is chosen, and a slice around that with some geodesic radius (distance along the surface) is computed. Points on the circle are selected using some angle step, and the normal at that point is determined. A super splash is when this process is repeated for several different radii. For each normal on the circle, additional angles between it and a coordinate system centred on the reference normal are computed. These angles and the angle around the circle are then mapped into a 3D space, where polygonal approximation is made, connecting each point with a straight line. Some additional computation is done to allow the encoded polygons to act as a hash. The method is quite complex.

Point Signatures are similar to the splash descriptor in the sense that they both sample points on a circle [8]. This descriptor again selects a reference normal, and has a specific radius. This time, the radius defines a sphere around the point. The intersection of the surface with the sphere is a 3D space curve. The orientation of the curve is defined by fitting a plane to it. The distances between the space curve and the fitted plane at sampled points define the signature of the reference point. These signatures can be compared by lining them up and checking whether the query falls within the tolerance band of

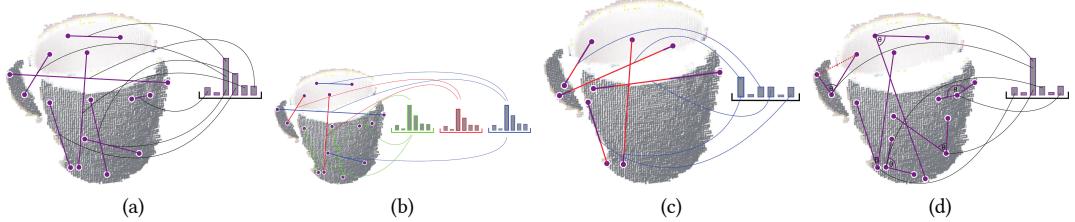


Figure 2: Examples of the measures used to construct the Ensemble of Shape Functions histograms of [35]. a) Distance between points. b) Whether the points are on or off the model, or mixed. c) Ratio of line segments on and off the surface of the model. d) Angle between pairs of lines.

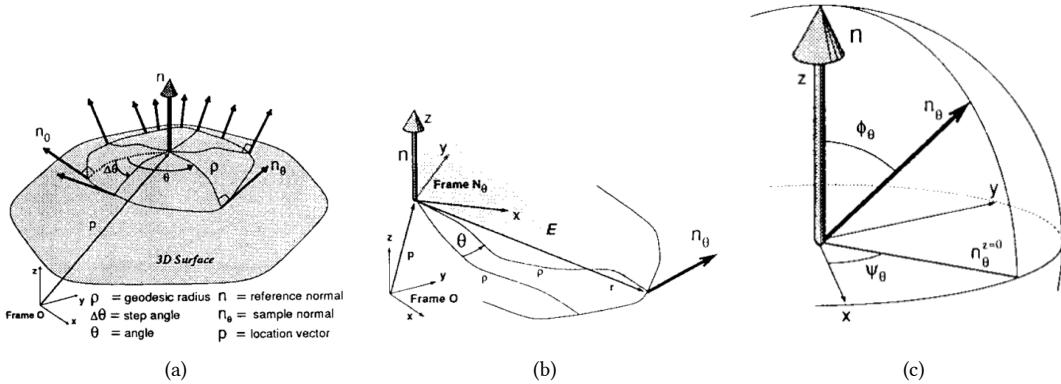


Figure 3: Splash descriptor [33]. a) shows the splash and normals around it. b) and c) show how the additional angles are defined.

previous signatures.

## 5 Point Matching

In [9], Chui and Rangarajan introduce an extension to ICP which allows for non-rigid registration and improved robustness to outliers. In contrast to ICP, their approach does not use the nearest-neighbour approach to define correspondence. Instead, they use an alternating algorithm similar to expectation maximisation. Annealing is used to prevent binary correspondences when the algorithm is not yet close to the solution — at the beginning there is a large search range for correspondences, which gradually shrinks as the temperature decreases.

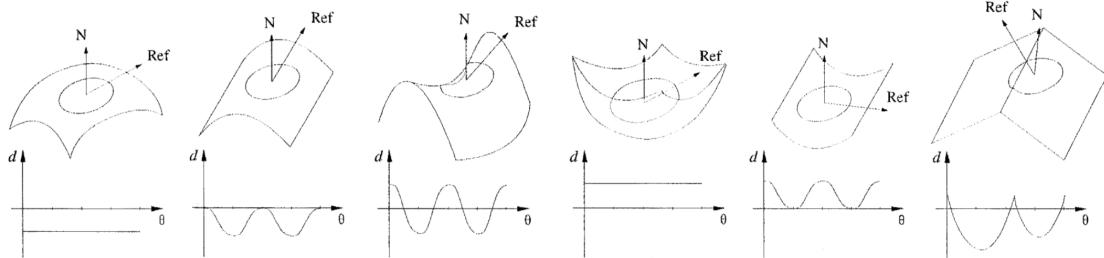


Figure 4: Examples of the point signature responses to different surfaces [8].  $d$  is the distance from the reference vector to the space curve defined by the intersection of the surface with a sphere centred at  $N$ . Ref rotates about  $N$ .



(a) Conformal factors. High value indicates high required deformation to sphere [4].

Figure 5: Model matching approaches

## 6 Model Matching

In [4], Chen et al. describe another approach to model matching using conformal factors. This technique uses ideas from conformal geometry, transforming the mesh of an object such that it has a uniform Gaussian curvature. Information is stored about how much deformation is needed locally to globally transform the object into a sphere – this is the conformal factor. The factor is based on a global computation on the whole mesh, as opposed to per-vertex computations of the Gaussian curvature, which makes it much smoother and appropriate for use in histograms. The histogram of a sample of the factors is used as a descriptor, and is pose invariant, as seen in Figure 5a. The authors say that it should be possible to use the approach in partial model matching.

## 7 Storing and Querying Descriptors

There are several techniques for storing and querying descriptors, mostly based on some form of tree. Recently, the k-d tree[5, 12] has been used for efficient approximate matching with either an error bound [2], where there is a bound placed on the error between the true nearest neighbour and the one found, or a time bound [3], where the search is stopped after examining a certain number of leaf nodes. Further improvements on the k-d tree are introduced in [31], where multiple randomised trees are used to optimise the search. A priority search tree algorithm is introduced in [22], which appears to be very effective. This may be the same one as in [21]. The algorithm in the last two papers has been integrated into PCL, which is useful.

The vocabulary tree [23] makes use of techniques from document search to index images. Using  $k$ -means clustering, construction stage creates a hierarchical quantisation of the image patch descriptors. In the query phase, descriptors are compared to the cluster centres, and go down the tree until a leaf is reached. The path through the tree is used as a scoring measure to present retrieval results.

Philbin et al. [25] show that flat (single-level)  $k$ -means clustering can be scaled to large vocabulary sizes if approximate nearest neighbour methods are used. Early systems for image retrieval used a flat clustering scheme, which could not scale to large vocabularies [32]. The paper also introduces a re-ranking method which uses spatial correspondences, which improves the retrieval quality.

Boiman et al. [6] introduce the Naive Bayes Nearest Neighbour (NBNN) classifier, which seems to be quite relevant. It uses nearest neighbour distances in the space of descriptors instead of images, computing “image-to-class” distances without quantising the descriptors. In general, quantisation allows for dimensionality reduction, at the expense of the discriminative power of descriptors. NBNN “can exploit the discriminative power of both (few) high and (many) low informative descriptors”. The problem here is that the classes must be known beforehand, and in our case we do not have that information. The local NBNN [19] does not do the search based on classes. Instead, all the descriptors are merged into a k-d tree on which approximate  $k$ -NN is run to find descriptors in the local region of a query descriptor. A distance to classes not present in the  $k$ -NN region is approximated by the distance to the  $k + 1$ th neighbour. Figure 6 shows a comparison of the two approaches.

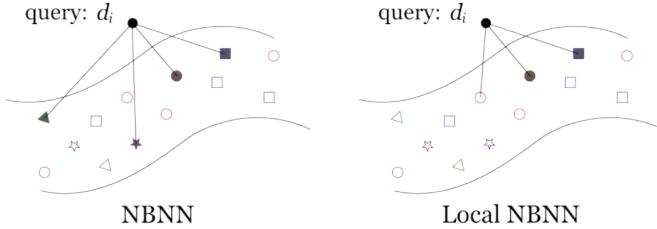


Figure 6: NBNN vs. local NBNN [19]. NBNN looks at the nearest neighbours in all classes, whereas local NBNN just looks at the local region, regardless of what classes are present in the database.

Funkhouser and Shilane present a method for querying a database of 3D objects represented by local shape features [14]. Partial matches (correspondences) are stored in a priority queue sorted by geometric deformation and the feature similarity. This means that only objects in the database with a high probability of being a match need to be processed.

Some work has been done on optimising the retrieval of relevant images by learning from user input [26]. When retrieved images are presented, the user ranks them in terms of relevance, and this rank is then used to improve the relevance of future searches.

## References

- [1] Aitor Aldoma et al. “CAD-model recognition and 6DOF pose estimation using 3D cues”. In: *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*. IEEE. 2011, pp. 585–592.
- [2] Sunil Arya et al. “An optimal algorithm for approximate nearest neighbor searching fixed dimensions”. In: *Journal of the ACM (JACM)* 45.6 (1998), pp. 891–923.
- [3] Jeffrey S Beis and David G Lowe. “Shape indexing using approximate nearest-neighbour search in high-dimensional spaces”. In: *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*. IEEE. 1997, pp. 1000–1006.
- [4] Mirela Ben-Chen and Craig Gotsman. “Characterizing Shape Using Conformal Factors.” In: *3DOR*. 2008, pp. 1–8.
- [5] Jon Louis Bentley. “Multidimensional binary search trees used for associative searching”. In: *Communications of the ACM* 18.9 (1975), pp. 509–517.
- [6] Oren Boiman, Eli Shechtman, and Michal Irani. “In defense of nearest-neighbor based image classification”. In: *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE. 2008, pp. 1–8.
- [7] Yizong Cheng. “Mean shift, mode seeking, and clustering”. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 17.8 (1995), pp. 790–799.
- [8] Chin Seng Chua and Ray Jarvis. “Point signatures: A new representation for 3d object recognition”. In: *International Journal of Computer Vision* 25.1 (1997), pp. 63–85.
- [9] Haili Chui and Anand Rangarajan. “A new point matching algorithm for non-rigid registration”. In: *Computer Vision and Image Understanding* 89.2 (2003), pp. 114–141.
- [10] Gregory Cipriano, George N Phillips, and Michael Gleicher. “Multi-scale surface descriptors”. In: *Visualization and Computer Graphics, IEEE Transactions on* 15.6 (2009), pp. 1201–1208.
- [11] Dorin Comaniciu and Peter Meer. “Mean shift: A robust approach toward feature space analysis”. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 24.5 (2002), pp. 603–619.
- [12] Jerome H Friedman, Jon Louis Bentley, and Raphael Ari Finkel. “An algorithm for finding best matches in logarithmic expected time”. In: *ACM Transactions on Mathematical Software (TOMS)* 3.3 (1977), pp. 209–226.
- [13] Keinosuke Fukunaga and Larry Hostetler. “The estimation of the gradient of a density function, with applications in pattern recognition”. In: *Information Theory, IEEE Transactions on* 21.1 (1975), pp. 32–40.
- [14] Thomas Funkhouser and Philip Shilane. “Partial matching of 3 D shapes with priority-driven search”. In: *ACM International Conference Proceeding Series*. Vol. 256. Citeseer. 2006, pp. 131–142.

- [15] Shengyin Gu et al. “Surface-histogram: A new shape descriptor for protein-protein docking”. In: *Proteins: Structure, Function, and Bioinformatics* 80.1 (2012), pp. 221–238.
- [16] Cheuk Yiu Ip et al. “Using shape distributions to compare solid models”. In: *Proceedings of the seventh ACM symposium on Solid modeling and applications*. ACM. 2002, pp. 273–280.
- [17] Tony Lindeberg. “Feature detection with automatic scale selection”. In: *International journal of computer vision* 30.2 (1998), pp. 79–116.
- [18] David G Lowe. “Distinctive image features from scale-invariant keypoints”. In: *International journal of computer vision* 60.2 (2004), pp. 91–110.
- [19] Sancho McCann and David G Lowe. “Local naive bayes nearest neighbor for image classification”. In: *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE. 2012, pp. 3650–3656.
- [20] Krystian Mikolajczyk and Cordelia Schmid. “Scale & affine invariant interest point detectors”. In: *International journal of computer vision* 60.1 (2004), pp. 63–86.
- [21] Marius Muja and David G Lowe. “Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration.” In: *VISAPP (1) 2* (2009).
- [22] Marius Muja and David G. Lowe. “Scalable Nearest Neighbor Algorithms for High Dimensional Data”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36.11 (2014), pp. 2227–40.
- [23] David Nister and Henrik Stewenius. “Scalable recognition with a vocabulary tree”. In: *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*. Vol. 2. IEEE. 2006, pp. 2161–2168.
- [24] Robert Osada et al. “Shape distributions”. In: *ACM Transactions on Graphics (TOG)* 21.4 (2002), pp. 807–832.
- [25] James Philbin et al. “Object retrieval with large vocabularies and fast spatial matching”. In: *Computer Vision and Pattern Recognition, 2007. CVPR’07. IEEE Conference on*. IEEE. 2007, pp. 1–8.
- [26] Yong Rui and Thomas Huang. “Optimizing learning in image retrieval”. In: *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*. Vol. 1. IEEE. 2000, pp. 236–243.
- [27] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. “Fast point feature histograms (FPFH) for 3D registration”. In: *Robotics and Automation, 2009. ICRA’09. IEEE International Conference on*. IEEE. 2009, pp. 3212–3217.
- [28] Radu Bogdan Rusu et al. “Fast 3d recognition and pose using the viewpoint feature histogram”. In: *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. IEEE. 2010, pp. 2155–2162.
- [29] Radu Bogdan Rusu et al. “Persistent point feature histograms for 3D point clouds”. In: *Proc 10th Int Conf Intel Autonomous Syst (IAS-10), Baden-Baden, Germany*. 2008, pp. 119–128.
- [30] Zujun Shentu et al. “Context shapes: Efficient complementary shape matching for protein–protein docking”. In: *Proteins: Structure, Function, and Bioinformatics* 70.3 (2008), pp. 1056–1073.
- [31] Chanop Silpa-Anan and Richard Hartley. “Optimised KD-trees for fast image descriptor matching”. In: *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE. 2008, pp. 1–8.
- [32] Josef Sivic and Andrew Zisserman. “Video Google: A text retrieval approach to object matching in videos”. In: *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE. 2003, pp. 1470–1477.
- [33] Fridtjof Stein and Gérard Medioni. “Structural indexing: Efficient 3-D object recognition”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14.2 (1992), pp. 125–145.
- [34] Luc Van Gool, Theo Moons, and Dorin Ungureanu. “Affine/photometric invariants for planar intensity patterns”. In: *Computer Vision—ECCV’96*. Springer, 1996, pp. 642–651.
- [35] Walter Wohlkinger and Markus Vincze. “Ensemble of shape functions for 3d object classification”. In: *Robotics and Biomimetics (ROBIO), 2011 IEEE International Conference on*. 2011, pp. 2987–2992.
- [36] Walter Wohlkinger and Markus Vincze. “Shape distributions on voxel surfaces for 3D object classification from depth images”. In: *Signal and Image Processing Applications (ICSIPA), 2011 IEEE International Conference on*. IEEE. 2011, pp. 115–120.