

Shape Recovery and Analysis of Large Screw Threads

C. Robertson and R. B. Fisher

Machine Vision Unit,
Institute for Perception, Action and Behaviour,
Division of Informatics, University of Edinburgh,
Edinburgh, EH1 2QL, UK
craigr@dai.ed.ac.uk

Abstract

In this work we have addressed the question of whether it is possible to extract parametric models of screw-threads from 3D range data. Results, example times for analysis and comparisons with known ground truths are given.

1 Introduction

Shape analysis of objects from range data (captured three dimensional co-ordinates of surface points) is a key problem in computer vision with several important applications in manufacturing, such as assembly, quality control and reverse-engineering.

The measurement of large screw threads is both a difficult and costly operation requiring the attention of skilled engineers. It is also vitally important to the engineering industry that these measurements are accurate and timely. In the oil industry, for example, threads of radius 50cm are not unusual and must be machined to high tolerances. When being fitted it is of great importance that manufacturing tolerances can be assessed before installation.

In this work we have examined whether it is possible to extract the parameters of screw threads from 3D scanned data as well as gaining some insight into the manufacturing tolerances used.

2 Data

2.1 Quality and Sampling

Data collection is performed using a moving-head, orthogonal laser stripe ranger which provides data in 0.1mm steps in the X-Y plane. Noise on the data is

around 15 microns standard deviation. In a single scan we can sample with a depth of field of around 50mm although when sampling over a volume we can achieve around 120° of a large radius bolt. Typical scanned data is shown in figure 3.

Artificial helix data has also been generated using a wide range of parameters with Gaussian noise added in order to test algorithm robustness.

2.2 Models

We have modelled the thread as its underlying mathematical structure allows, that is as two co-axial helices with the same progression parameter but different radii. The helical parameters are shown in figure 1. In the canonical orientation (axis $\mathbf{n} = (0,0,1)$) the helix is given parametrically by:

$$x = r.\cos(t) \quad (1)$$

$$y = r.\sin(t) \quad (2)$$

$$z = b.t \quad (3)$$

where r is the helix radius and b is the helical progression. In our case this is expressed as movement in the direction of \mathbf{n} in millimetres per radian.

In an acanonical orientation, if the axis \mathbf{n} is computed, then the helix can be rotated into the canonical orientation.

3 Algorithm

The method is split into the following parts: data collection, rotation to canonical alignment, radius assessment and residual calculations, progression analysis. Each is discussed below and the processing pipeline is shown in figure 2, below.

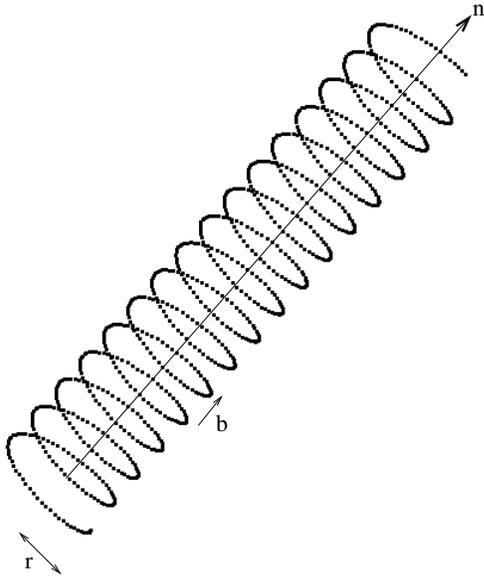


Figure 1. Helix Parameters

3.1 Scanning

Data scanning was performed using a 3D Scanners flat bed laser range sensor with a 5 degrees of freedom scanning head. In our case, because we were using orthogonal (volume) scanning, only 3 degrees of freedom were used (X, Y and Z). Artificial data was generated with a random axis rotation with Gaussian noise added with various kernel sizes. The minimum size was $0.2mm$ which is designed to be much larger than the accuracy with which we can detect the peaks and troughs in the scanned data.

3.2 Rotation to Canonical Alignment

In order to perform the re-alignment, the dual helix axis has first to be found. This is a two stage process:

1. *Stage 1 - Peak and trough finding.* First we have to find the maximal and minimal points of the screw thread lying on the inner (minor) radius and the outer (major) radius. This de-couples the helices from the rest of the screw. We do this by performing a nearest-neighbour analysis. Because of the screw structure, the points which have the highest number of neighbours are those that are at turning points in the data (figure 4). An implicit assumption is that the screw is scanned in a direction that is approximately orthogonal to the direction of the screw axis. This ensures that both the top and bottom of the screw cut is visible.

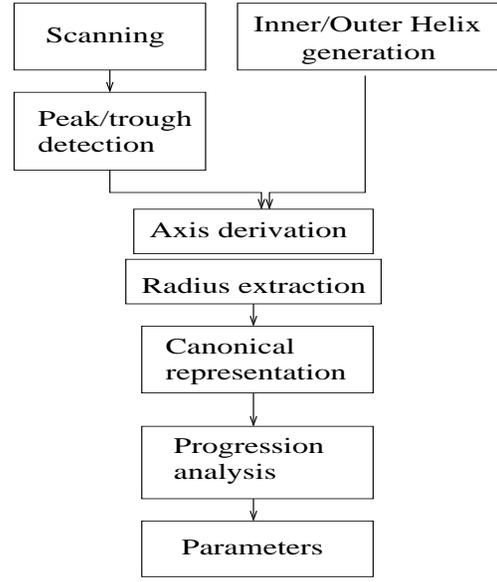


Figure 2. Screw Thread Detection Processing Pipeline

2. *Stage 2 - Inner and outer helix separation and axis fitting.* This optimization is performed using a genetic algorithm. The data taken from the peak and trough finder is treated as belonging to two co-axial cylinders of differing radii. The function on which the optimization is performed is the least-squares fitting of the two cylinders. The chromosome used for this fitting is as follows: $\langle Nx, Ny, Nz, Px, Py, Pz, R1, R2 \rangle$ where the first 3 terms are the axis normal direction, the second three terms are a point in 3D and the final two are the two different cylinder radii.

Automatic segmentation is also made implicit in the evaluation function:

```

for all points, i
{
  x = points[i][1]
  y = points[i][2]
  z = points[i][3]
  d1 = distance(X[1] - X[6], X[7], x,y,z)
  d2 = distance(X[1] - X[6], X[8], x,y,z)
  fitness=0.0
  if(d1 < d2)
  {
    fitness+=d1
  }
  else

```

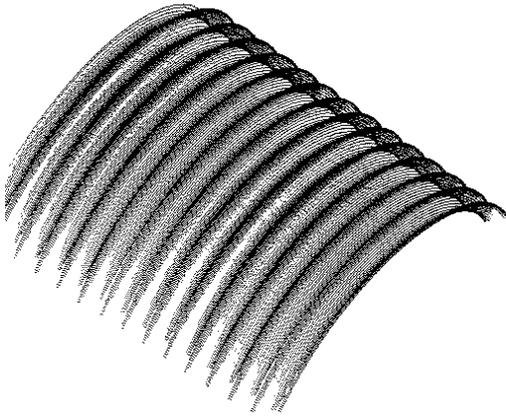


Figure 3. Typical Scanned Data

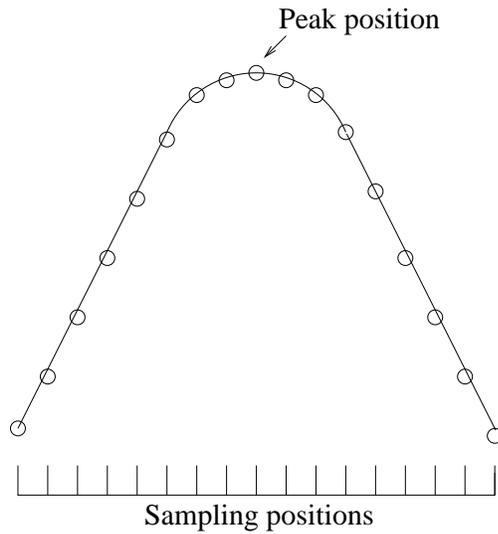


Figure 4. Screw Cross-section

```
{
  fitness+=d2
}
```

Note that we are minimizing the fitness function. The function *distance()* returns the distance from the passed point to the cylinder with the passed parameters. The distance to each cylinder is assessed and the lowest value is taken.

3.3 Radius Analysis

Once the data axis has been found we can estimate the inner and outer radii at various points along it to

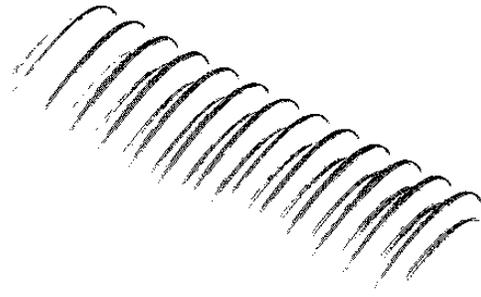


Figure 5. End of Stage 1, Peaks and Troughs Detected

ensure that the screw cut is consistent. This gives us both radial measurements (with standard deviation). Note that we could also deal with eccentric helices by using the guaranteed ellipse fitter [1] on the data projection onto the base plane. An example of the inner radius estimate histogram is shown in figure 6, below.

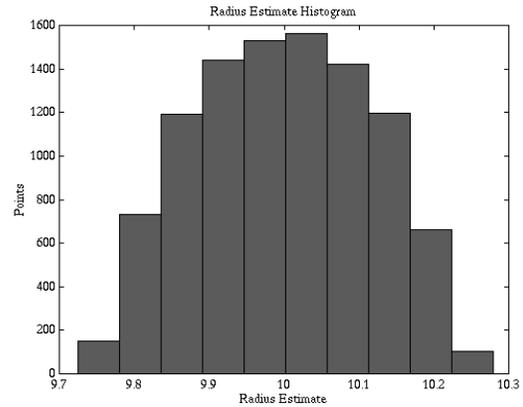


Figure 6. Radius Estimates Histogram

3.4 Progression Analysis

We can estimate the progression rate by finding pairs of points which lie approximately in alignment with the helix pair axis direction, but are separated in the Z direction, as shown in figure 7. Thus, we require a distance, $d = kb$, such that for a point P_0 we can find another point P_1 satisfying :

$$P_k = P_0 + kb\mathbf{n} \quad (4)$$

where b is the progression rate and \mathbf{n} is the axis normal for the two helices and k is the integer multiple of the progression step to move to the next thread.

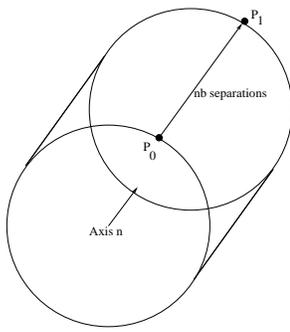


Figure 7. Progression Rate Finding

Plotting d gives a graph of progression distances in Z , as shown in figure 8. Finding the progression rate is then simply a matter of grouping and averaging the distances between groups. Computing the standard deviation around the fit will also give a measure of consistency in the cutting of the thread.

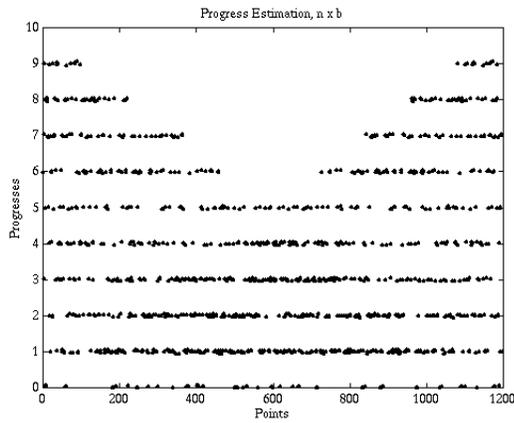


Figure 8. Progression Rate Estimation

4 Results

Results are presented for both real and generated data together with example timings.

4.1 Parameterisations

Generated Data

For 100 sets of randomly generated pairs of helix data, representing screw threads of a wide range of radii and progression rate and at random orientations we found that radii were found to $0.01mm$ ($< 0.1\%$) even when subjected to Gaussian noise on all three components of values between $0.2mm$ and $0.4mm$. This level

of error is higher than we would realistically expect, however, by at least an order of magnitude.

The progression rate, however, was less accurate due to the smaller number of data points. This had an average estimate error of $0.02mm/radian$ (1%), which is satisfactory for this approach.

Scanned Data

Several large screw threads were scanned, each of the same radii and for which ground truth was known. In each case the amount of data collected from one orthogonal scan only was considered and this represented around 30° of the screw in each case. The scanned area ($60mm$ by $40mm$) generates around 100,000 points when scanned at $100\text{ points}/mm^2$ in the XY plane.

An example graph of the convergence of the least squares error of the GA is shown in figure 9. An exam-

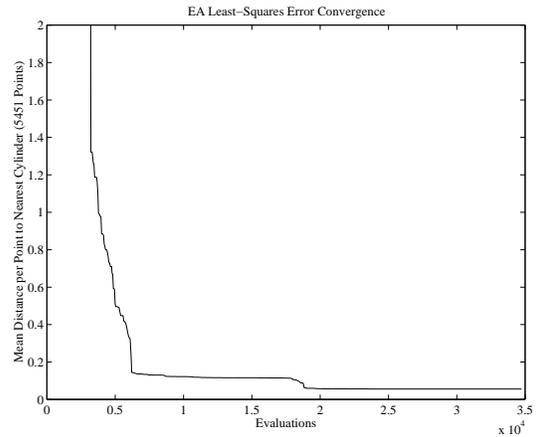


Figure 9. GA: Error Convergence

ple of the convergence of the inner/outer radii is shown in figure 10. Note that this convergence is from a set of random starting values.

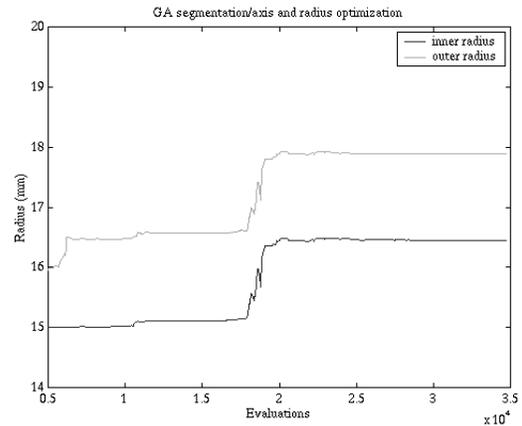


Figure 10. GA: Convergence of Radii

The unit axis value convergence is shown in figure 11. Note that one value changes very quickly from -1 to 1. This is a boundary mutation operation that is a standard part of the GA operator set.

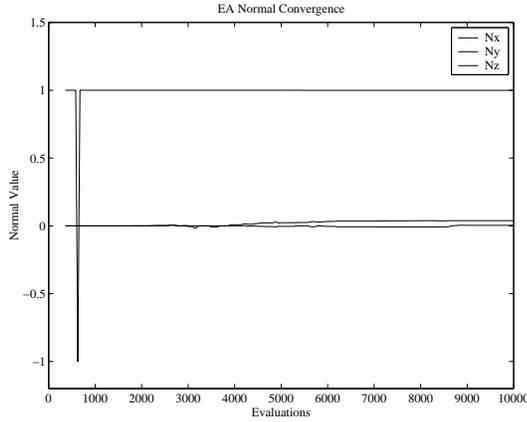


Figure 11. GA: Axis Convergence

Four scanned data sets were analysed and their parameters were compared to their measured ground-truth. Two tables of results are shown in table 1 and table 2, below.

Screw	Absolute inner/ radius error/mm	Absolute outer/ radius error/mm
1	0.09	0.07
2	0.09	0.07
3	0.07	0.04
4	0.11	0.10

Table 1. Radius Estimate Errors for Large Screw Threads (True values 16.36mm and 17.77mm)

Screw	Absolute progression error in mm/radian
1	0.0010
2	0.0009
3	0.0012
4	0.0021

Table 2. Progression Estimate Errors for Large Screw Threads (True Value = 0.37 mm/radian)

4.2 Example Timings

The timings presented here, with the exception of scanning, are using a 700MHz Athlon PC.

Scanning

Scanning of large threaded areas, say $60mm \times 60mm$ takes approximately 10 minutes at a resolution of $0.1mm \times 0.1mm$.

Rotation to Canonical Alignment (Axis Finding)

The most CPU intensive part of the analysis is the peak and trough finding which has to perform an $O(n^2)$ nearest neighbour calculation. With around 100,000 points per dataset this can take 90 minutes.

The axis finding GA takes around 25 minutes when using a 10,000 point peak/trough set and a random starting position on the solution manifold. If good domain constraints are used, utilising *a priori* knowledge then this time is heavily reduced. A search population of 200 was used, together with a reference population of 100 chromosomes. The strategy was as described in [2].

Radius Assessment and Residual Calculations

The radius assessment and residual calculations take a very short time, less than 10 seconds for a 100,000 point dataset.

Progression Analysis

Progression analysis also takes less than 10 seconds for a 100,000 point dataset.

5 Conclusions

In this paper we have addressed the question of whether it is possible to measure large screw threads in an analytical, rather than a physical way by direct measurement. We have shown that to a great extent this is not only possible but can be done robustly to give not only the parameters of the thread but some indication of manufacturing quality and stability. These are two factors that make this technique particularly useful to industry.

In our results, as expected, we found that randomly generated data gave roughly comparable results to real data. The inner and outer radius finding is especially good even when using fairly poor data (machined metal does not scan particularly well) so long as the scan is performed across the thread, rather than down it.

It is entirely possible that the method here could be used equally well in a stand-alone scanning unit.

Acknowledgements

The work presented in this paper was funded by UK EPSRC grant GR/M97138.

References

- [1] A. W. Fitzgibbon, M. Pilu, R. B. Fisher, "Direct Least Squares Fitting of Ellipses", *IEEE Trans. Pat. Anal. and Mach. Intel.*, 21(5), pp476–480, 1999.
- [2] C. Robertson, R.B. Fisher, D. Corne, N. Werghi, A.P. Ashbrook, "Investigating Evolutionary Optimisation of Constrained Functions to Capture Descriptions from Range Data", *Advances in Soft Computing - Engineering Design and Manufacturing*. R. Roy, T. Furuhashi and P. K. Chawdhry (Eds), Springer, 1999, pp 455–466.