Predicting Features in Complex 3D Surfaces Using a Point Series Representation: A Case Study in Sheet Metal Forming

Subhieh El-Salhi, Frans Coenen, Clare Dixon, and Muhammad Sulaiman Khan

University of Liverpool, Department of Computer Science, United Kingdom
{hsselsal,cldixon,mskhan}@liv.ac.uk

Abstract. This paper presents an integrated framework for learning to predict geometry related features with respect to 3D surfaces. The idea is to use a training set of known prediction values to create a model founded on local 3D geometries associated with a given surfaces so that predictions with respect to a new "unseen" surfaces can be made. The local geometries are represented using point series curves. Two variations are proposed: (i) *discretised* and (ii) *real number*. To act as a focus for the work a sheet metal forming application is considered where we wish to predict the errors that are introduced as a result of applying a forming process. Given a desired surface T, the surface T' actually produced as a result of the sheet metal forming process is affected by a phenomena called *Springback* (the feature we wish to predict). The proposed process has been evaluated using two flat-topped pyramid shapes and by considering a variety of parameter settings. Excellent results have been obtained in terms of accuracy and Area Under ROC Curve (AUC).

Keywords: 3D Surface Representation, Point series curves, Classification.

1 Introduction

Pattern mining has long been studied within the field of knowledge discovery in data and machine learning. Pattern mining in the context of tabular data is well understood (see for example work on frequent item set mining). With respect to many other forms of more complex data, pattern mining remains a subject of research. The fundamental challenge is to represent such complex data in a way that facilitates meaningful pattern identification. The focus for the work described in this paper is three dimensional (3D) surfaces. More specifically we wish to identify patterns in such surfaces that are indicative of some set of feature values we wish to predict. The intention is that, once identified, these patterns can be used for prediction purposes with respect to new "unseen" 3D surfaces. The idea is to represent specific local geometries located within 3D surfaces so that we can attempt to associate specific feature values with these geometries. To this end a prediction framework is proposed whereby local geometries are represented in terms of linearisations to form a collection of point series (curves). Two variations of the linearisation are considered, a discretised linearisation and a real

M. Yao et al. (Eds.): ADMA 2013, Part I, LNAI 8346, pp. 505–516, 2013.
© Springer-Verlag Berlin Heidelberg 2013

valued linearisation, and a number of different local geometry "sizes". Using an appropriately defined training set prediction values, associated with some feature of interest, can be related to each linearisation. Given a new "unseen" 3D surface this can be decomposed into a collection of linearisations similar to those used with respect to the training set. By matching the new linearisations associated with the new surface to the previously generated linearisations, which have known feature values associated with them, these values can be used as predictor values with respect to the new shape.

The exemplar application, and that used to illustrate the work described in this paper, is sheet metal forming. In sheet metal forming the forming process takes as input a specification of a 3D surface to be manufactured, a shape T, to which a forming processes is applied to produce a shape T'. However, as a result of application of the process various deformations are introduced, called *springback*. As a consequence the manufactured shape T' is not equivalent to the specified shape T. Springback is caused by a number of factors of which the most significant is the geometry of the intended shape [2]. Further, springback is not distributed evenly over a given shape, in practice springback is more significant with respect to some geometries than others. Springback is therefore correlated to the nature of local geometries within the specified shape. If we can predict springback we can apply a correction to the specification T so as to minimise the springback effect.

Thus the contributions of this paper are as follows:

1. A method for representing local 3D geometries using point series.
2. A mechanism for feature value prediction with respect to local geometries present within 3D surfaces.
3. An interesting case study illustrating the significance of the proposed process.

The rest of the paper is organised as follows. In section 2 a brief overview of related work is presented. Section 3 introduces the prediction framework including the generation of the proposed point series representations. The evaluation of the proposed prediction framework, using manufactured surfaces describing flat topped pyramid shapes, is presented in Section 4 using a variety of parameters. Some conclusions are then presented in Section 5.

2 Overview of Related Work

The work described in this paper, although generally applicable, is directed at sheet metal forming as widely used in the aircraft and automotive parts manufacturing industries. A specific issue in sheet metal forming is the springback phenomenon, the elastic deformation that occurs as a result of the application of the manufacturing process. Generally speaking the nature of any resulting springback is influenced by: (i) the manufacturing parameters used and (ii) the material properties [6,15,13]. Substantial works have been conducted to characterise, analyse and predict the springback. To this end the Finite Element Method (FEM) has been extensively used for springback prediction purposes [4,14,19]. The FEM provides a simulation environment that is flexible (parameters can be easily modified) but at the same time complex. However, the application of FEM is a time consuming task [7,17,6]. Furthermore, FEM has been found to be an

inaccurate prediction method due to the use of simplification assumptions with respect to the required integration calculation [3,4,15]. An alternative approach, and that advocated in this paper is to use some form of machine learning to build a predictor. More specifically to generate a generic classifier that takes as input definitions of local geometries contained on the part to be manufactured and predicts springback. To the best knowledge of the authors there is no reported work on the application of classification techniques for the purpose of predicting springback in sheet metal forming.

3 The Prediction Framework

The input to the proposed prediction framework is a grid describing some 3D shape of interest. Each grid square is defined by its centre point, the *grid point*, which is in turn defined by an *x*-*y* coordinate pair. Each grid point also has a *z* (height) value and, when used as a training set, a value associated with some feature of interest. In the case of the sheet metal forming application this will be a springback value. Thus the grid represents a mesh describing some 3D surface of interest. Using this input we wish to build a model of the 3D surface which can be used to predict values associated with new "unseen" 3D surfaces. Note that the number of grid squares required to represent a given shape will depend on the grid size d. Fewer grid squares will be generated if a larger d value is used.

Given a grid representation G, a collection of local geometries can be defined (one per grid square). Two alternative proposed point series representations are described in Sub-section 3.1 below. Once a collection of point series curves have been generated, each with a feature value of interest associated with it (a prediction value), this model can be used for prediction purposes. There are a number of mechanism where by this may be achieved; however, in the context if this paper a K-Nearest Neigbour (KNN) approach is advocated that uses the *warping distance* between local geometry curves to make predictions. This matching process is described further in Sub-section 3.2.

3.1 Point Series Representation

There are various ways that 3D local geometries can be defined. Earlier work by the authors considered a Local Binary Pattern (LBP) based representation referred to as the Local Geometry Matrix (LGM) representation [5,11,12]. In the context of sheet metal forming the authors have also considered a "distance from edge" measure (unpublished), referred to as the Local Distance Measure (LDM) representation. In this paper a point series representation is considered. The basic idea is to describe each local 3D geometry surrounding a grid point p_i using a linearisation of space. There are four issues to be considered in this respect:

1. What is the nature of the neighbourhood to be considered.
2. How best to conduct the desired linearisation.
3. How many points should we include in our point series.
4. What is the nature of the value represented by each point.

In terms of the nature of the neighbourhood this is partly related to the adopted grid size d, adoption of a large value for d will dictate larger grid squares and consequently larger neighbourhoods. The size of a neighbourhood can be simply described in terms of a Region Of Interest (ROI) surrounding each point p_i defined in terms of a $n \times n$ block of grid squares centred on p_i (note that to ensure that the ROI is symmetric about p_i the value for n should be an odd number). What the value for n should be is a matter for consideration and may also, at least to an extent, be application dependent. For the linearisation there are a number of "space filling curve" formats that could have been adopted (for example a Peano curve [16] or a Hilbert curve [8]). However, given the nature of our ROIs (see below) a straightforward spiral linearisation was adopted as this fits well with the proposed $n \times n$ ROI definition. With respect to the number of points to be considered we can include all points covered by a linearisation or a selection of "key" points. If we include all points there will be $n^2 - 1$ points per linearisation. Given a grid comprised of many grid squares this will result in a large number of point series (one per grid square) which in turn might mean that the size (length) of each point series becomes significant (from a time complexity perspective). An alternative is to consider only "corner points" and "mid-way points" in which case we will have $((n-1)/2) \times 8$ points per linearisation instead of $n^2 - 1$. Finally, with respect to the value to be represented by each linearisation point the idea is to use the difference in the z coordinates between each neighbourhood point and the point p_i as this will clearly capture variations in local geometries. The issue is weather to use real δz values or discretised values. In the case of the LGM and LDM representations considered previously by the authors (see above) discretised values were used. This was because the classification processes adopted with respect to these representations required discrete values. However, real values may produce a better result. Thus two variations off the proposed linearisation were considered: (i) discretised linearisation and (ii) real number linearisation. Figure 1(a) shows an example linearisation using a 5×5 neighbourhood and key points instead of all points. Figure 1(b) shows the spiral linearisation and Figure 1(c) the resulting point series.

Using the above process we can create a model of a particular 3D surface application domain which can be used for prediction purposes. So that the model is as comprehensive as possible the training data should cover a wide diversity of local geometries, ideally all the potential local geometries that can occur with respect to a particular application domain.

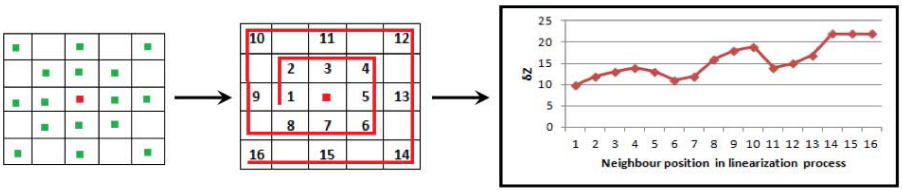

Fig. 1. Example spiral linearisation for a 5×5 neighbourhood

3.2 Prediction

As noted above a KNN classification systems is suggested to carry out the desired prediction. Given a new curve to be classified, the k most similar pre-labeled curves were identified and the most similar selected according to the warping distance between two curves. The warping distance was obtained using the established Dynamic Time Warping (DTW) process [10,18]. DTW operates as follows. Given two point series $A = \{a_1, a_2, \cdots, a_n\}$ and $B = \{b_1, b_2, \cdots, b_m\}$, we create a matrix R that has dimension $|A| \times |B| = n \times m$ is defined. The value at each matrix element $\langle i, j \rangle$ (where $0 \leq i < n$ and $0 \leq j < m$) is the the Euclidean distance between points a_i and b_j when the two curves are considered in term of a 2D reference plane. Once the matrix elements have been computed the *warping path* W is identified ($W \subset elements\ in\ R$). This is the path from the bottom left corner of the matrix to the top right corner that links the matrix elements with lowest values. The most direct path will be along the leading diagonal, this will occur when curves A and B are identical. The length of the warping path, W_{dist}, is the accumulated sum of the matrix element values contained in W:

$$W_{dist} = \sum_{k=1}^{k=|W|} W_k \tag{1}$$

Given two identical curves W_{dist} will be 0.

4 Evaluation

The evaluation of the proposed prediction framework is considered in this section. To this end real data was obtained using an Asymmetric Incremental Sheet Forming (AISF) process, a sheet metal forming process used in industry [9]. Two sample surfaces (shapes) were considered referred to as the Gonzalo and Modified surfaces, both described flat topped pyramid shapes. Each was manufactured four time, twice using steel and twice using titanium. Thus eight different data sets (comprised of both before and after clouds) were available for experimentation: (i) Gonzalo Steel 1 (GS1), (ii) Gonzalo Steel 2 (GS2), (iii) Modified Steel 1 (MS1), (iv) Modified Steel 2 (MS2), (v) Gonzalo Titanium 1 (GT1), (vi) Gonzalo Titanium 2 (GT2), (vii) Modified Titanium 1 (MT1), (viii) Modified Titanium 2 (MT2). Each data set comprised two coordinate clouds, a before cloud C_{in} and an after cloud C_{out}. The pre-processing of this data is described in Sub-section 4.1 below.

The evaluation was directed at investigating the following:

1. The effect of variations in the value of d (the grid size).
2. The effect of using different sized neighbourhoods.
3. A comparison with respect to using all linearisation points versus only key points.
4. The generalisation of the proposed approach (can we build a generally applicable classifier using our proposed linearisation).

Each of these is considered in further detail in Sub-sections 4.2 to 4.5 below. The evaluation was conducted in sequence by considering pairs of data sets, the best parameter

settings identified with respect to one investigation were adopted for use in following investigations. The metrics used for evaluation purposes were accuracy and AUC. A tolerance of 0.08, as suggested by BS ISO 2005 [1], was used.

4.1 Training Data Generation

As noted above the training data was derived using before and after point clouds (C_{in} and C_{out}) produced using AISF sheet metal forming. The C_{in} cloud was obtained from a CAD system and was used to describe the desired shape T. The C_{out} cloud, used to describe the produced shape T', was obtained using a GOM (Gesellschaft fur Optische Messtechnik) optical measuring tool. During pre-processing both clouds were translated into the desired a grid representation (using the same d value) to give two grids, G_{in} and G_{out} respectively. The z value associated with each p_i was obtained by averaging the z coordinates for all the points located within that grid square. The springback (error) associated with each p_i was obtained by comparing the corresponding points in G_{in} and G_{out}. The springback value in each case was obtained by calculating the distance along the normal from each point p_i in C_{in} to where it cut the C_{out} surface.

4.2 Grid Size

To determine the effete of the grid size, a range of grid sizes, $d = \{2.5, 5, 10, 15, 20\}$ mm, were considered together with a 3×3 neighbourhood using the key point linearisation. The results are shown in the Tables 1 (best result for each dataset indicated in bold font). From the table it can be seen that excellent accuracy and AUC values were recorded regardless of the shape and manufacturing material used. From the tables it can also be observed that, in general, as the grid size increases the accuracy and AUC values start to decrease, it is conjectured that this is because of the increasing coarseness of the representation. This is more evident with respect to the AUC values because of the unbalanced nature of the input data.

Table 1. The results for 3×3 neighbourhood using *key* point representation technique (Total number of key points = 8 points)

	d=2.5		d=5		d=10		d=15		d=20	
	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC
GSV1	0.97	0.96	0.98	0.97	0.98	0.96	0.97	0.95	0.90	0.82
GSV2	**0.99**	0.94	0.98	0.89	0.97	0.84	0.96	0.64	0.96	0.78
GTV1	**0.99**	**0.97**	**1.00**	**1.00**	**0.99**	**0.98**	0.94	0.76	0.93	0.72
GTV2	**0.99**	0.96	0.99	0.99	**0.99**	0.97	**0.98**	0.93	0.96	**0.96**
MSV1	0.97	0.92	0.97	0.92	0.98	0.97	0.97	0.94	**0.97**	0.87
MSV2	0.96	0.94	0.98	0.97	0.98	0.93	0.97	0.92	0.92	0.71
MTV1	0.98	0.96	0.98	0.97	0.98	0.94	0.97	**0.96**	0.90	0.81
MTV2	0.97	0.96	0.96	0.94	0.93	0.90	**0.98**	0.94	0.82	0.73

4.3 Neighbourhood Size

To determine the effect of different $n \times n$ neighbourhood sizes three values of n were considered, $\{3,5,7\}$. Tables 2 and 3 show the results obtained using 5×5 and 7×7 neighbourhood sizes. Comparing these results with those presented above in Table 1 (3×3) it can be seen there there is no discernible difference between the different neighbourhood sizes although it can be noted that grid size ceases to have an impact as neighbourhood sizes increase.

Table 2. The results for 5×5 neighbourhood using *key* point representation technique (total number of key points = 16 points)

	d=2.5		d=5		d=10		d=15		d=20	
	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC
GSV1	0.97	**0.96**	0.97	0.95	**0.98**	**0.97**	**1.00**	**1.00**	0.84	0.80
GSV2	**0.99**	0.94	0.97	0.89	0.96	0.75	0.94	0.64	0.92	0.73
GTV1	**0.99**	**0.96**	**0.99**	**0.99**	0.94	0.89	0.94	0.70	0.91	0.74
GTV2	**0.99**	**0.96**	**0.99**	0.98	0.96	0.85	0.95	0.90	**0.99**	**0.99**
MSV1	0.96	0.91	0.96	0.92	0.96	0.92	0.98	0.97	0.91	0.62
MSV2	0.96	0.94	0.96	0.94	0.97	0.91	0.97	0.93	0.92	0.64
MTV1	0.98	**0.96**	0.98	0.96	0.96	0.89	0.97	0.95	0.84	0.70
MTV2	0.97	**0.96**	0.96	0.94	0.93	0.92	0.96	0.93	**0.99**	0.98

Table 3. The results for 7×7 neighbourhood using *key* point representation technique (total number of key points = 24 points)

	d=2.5		d=5		d=10		d=15		d=20	
	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC
GSV1	0.98	**0.97**	**0.99**	0.98	0.94	0.92	0.87	0.70	0.78	0.33
GSV2	**0.99**	0.96	0.98	0.93	0.94	0.85	0.89	0.65	0.67	0.50
GTV1	0.98	0.93	**0.99**	**0.99**	0.87	0.77	0.79	0.57	**0.81**	**0.89**
GTV2	**0.99**	0.95	0.98	0.96	0.88	0.72	0.83	0.75	0.48	0.19
MSV1	0.96	0.92	0.97	0.95	**0.99**	**0.99**	0.89	0.81	0.75	0.67
MSV2	0.97	0.95	0.97	0.97	0.96	0.95	0.92	0.86	0.67	0.50
MTV1	0.98	0.95	0.98	0.97	0.92	0.72	**0.97**	**1.00**	0.70	0.47
MTV2	0.97	0.94	0.96	0.93	0.97	0.95	0.92	0.74	0.56	0.25

4.4 All Points versus Key Points

This sub-section compares the operation of the all points linearisation with the key points linearisation using different grid size $d = \{2.5,5,10,15,20\}$. Table 4 presents the results obtained using a 5×5 neighbourhood coupled with the all point linearisation which can be compared with the results presented in Table 2 which shows the results of using a 5×5 neighbourhood and the key point linearisation. From the table it can be observed that there is no performance difference between the two linearisations. Similar results were produced using a 7×7 neighbourhood, Note that in the case of a

Table 4. The results for 5×5 neighbourhood using *all* point representation technique (total number of key points = 24 points)

	d=2.5		d=5		d=10		d=15		d=20	
	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC
GSV1	0.96	0.95	0.97	0.95	**0.98**	**0.97**	**1.00**	**1.00**	0.84	0.80
GSV2	**0.99**	0.94	0.98	0.90	0.96	0.75	0.94	0.64	0.92	0.73
GTV1	**0.99**	0.96	**0.99**	**0.99**	0.94	0.88	0.94	0.70	0.91	0.74
GTV2	**0.99**	0.96	**0.99**	0.98	0.96	0.85	0.95	0.90	0.98	**0.99**
MSV1	0.97	0.91	0.96	0.93	0.96	0.92	0.98	0.97	0.93	0.64
MSV2	0.96	0.94	0.96	0.94	0.96	0.92	0.98	0.96	0.90	0.64
MTV1	**0.99**	**0.97**	0.98	0.95	0.96	0.91	0.97	0.96	0.83	0.67
MTV2	0.98	**0.97**	0.96	0.94	0.93	0.92	0.98	0.96	**0.99**	0.98

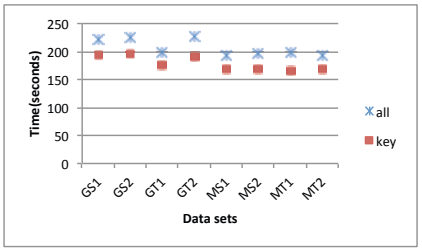

Fig. 2. The run time (s) for *all* vs *key* point representation, 5×5 neighbourhood, $d = 2.5$ mm

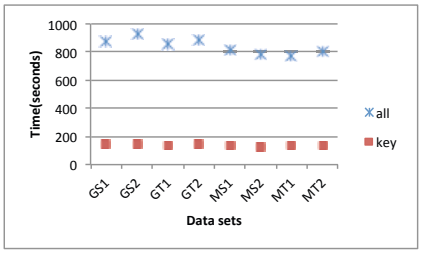

Fig. 3. The run time (s) for *all* vs *key* point representation, 7×7 neighbourhood, $d = 2.5$ mm

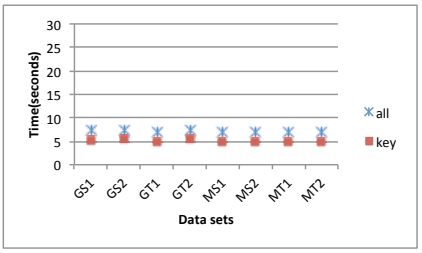

Fig. 4. The run time (is) for *all* vs *key* point representation, 5×5 neighbourhood, $d = 5$ mm

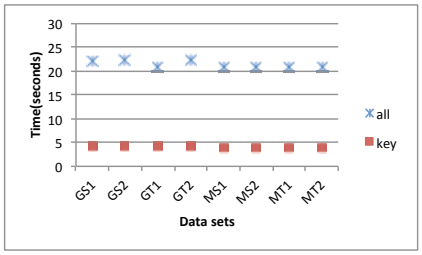

Fig. 5. The run time (s) for *all* vs *key* point representation, 7×7 neighbourhood, $d = 5$ mm

3×3 neighbourhood there is no difference between the key point and the all point linearisation.Figures 2 to 11 indicate the recorded run times using both representations and different grid size. The charts indicate that the key point linearisation is more efficient than the all point representation especially with respect to large grid sizes.

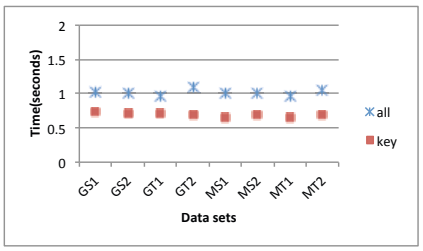

Fig. 6. The run time (s) for *all* vs *key* point representation, 5×5 neighbourhood, $d = 10$ mm

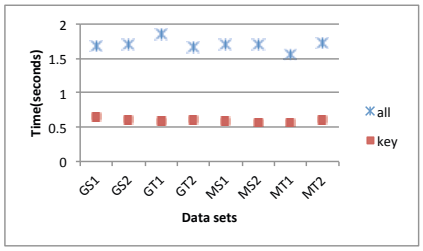

Fig. 7. The run time (s) for *all* vs *key* point representation, 7×7 neighbourhood, $d = 10$ mm

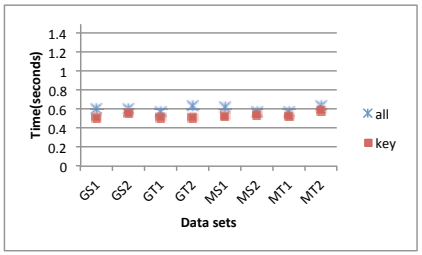

Fig. 8. The run time (s) for *all* vs *key* point representation, 5×5 neighbourhood, $d = 15$ mm

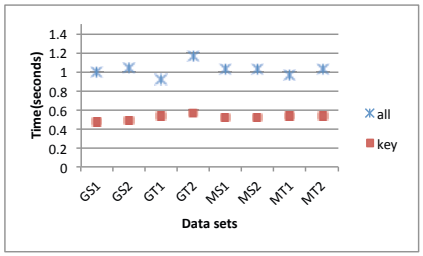

Fig. 9. The run time (s) for *all* vs *key* point representation, 7×7 neighbourhood, $d = 15$ mm

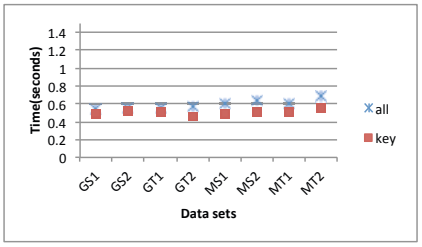

Fig. 10. The run time (s) for *all* vs *key* point representation, 5×5 neighbourhood, $d = 20$ mm

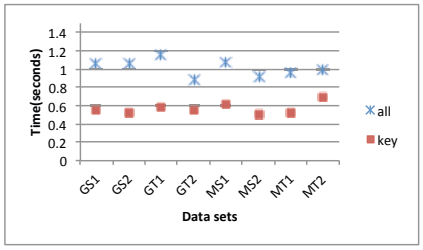

Fig. 11. The run time (s) for *all* vs *key* point representation, 7×7 neighbourhood, $d = 20$ mm

4.5 Generalisation

This section presents the results obtained from training a classifier on one shape and testing it on another. The main goal was determine whether it was possible to generate a generally applicable classifier if it was provided with a suitable shape to train on. From earlier experiments it was noted that that lower grid sizes produced better results, hence for this set of experiments $d = 5$ was used. The operation of the proposed point series representation was also compared with the LGM and LDM representations proposed previously by the authors [5]. Table 5 presents the results obtained in terms of AUC values (best results on bold). From the table it can be seen that using the proposed representation a best AUC value of 1.00 could be obtained. In terms of the the previous

Table 5. AUC results for the generic classifier

			Train							
			GSV1	GSV2	GTV1	GTV2	MSV1	MSV2	MTV1	MTV2
Test	GSV1	Point series		**0.97**	**0.94**	**0.93**	**0.96**	**0.91**	**0.96**	**0.98**
		LGM		0.66	0.59	0.70	0.44	0.52	0.48	0.52
		LDM		0.52	0.50	0.50	0.51	0.50	0.52	0.47
		LGM + LDM		0.94	0.76	0.81	0.80	0.89	0.75	0.70
	GSV2	Point series	**0.99**		**0.99**	**0.92**	**1.00**	**0.96**	**1.00**	**0.99**
		LGM	0.62		0.68	0.74	0.60	0.53	0.67	0.61
		LDM	0.50		0.50	0.50	0.50	0.50	0.50	0.50
		LGM + LDM	0.72		0.78	0.83	0.81	0.90	0.78	0.74
	GTV1	Point series	**0.84**	0.87		**0.95**	0.64	0.66	**0.94**	**0.94**
		LGM	0.65	0.74		0.75	0.69	0.67	0.69	0.67
		LDM	0.49	0.61		0.53	0.41	0.41	0.41	0.40
		LGM + LDM	0.70	**0.89**		0.80	**0.80**	**0.89**	0.73	0.72
	GTV2	Point series	**0.81**	0.91	**0.98**		0.65	0.61	**0.99**	**0.95**
		LGM	0.66	0.81	0.72		0.70	0.63	0.68	0.65
		LDM	0.50	0.50	0.50		0.50	0.50	0.50	0.50
		LGM + LDM	0.74	**0.97**	0.79		**0.83**	**0.93**	0.79	0.74
	MSV1	Point series	**0.98**	**0.99**	**0.99**	**0.98**		**0.95**	**0.98**	**0.97**
		LGM	0.61	0.57	0.70	0.76		0.82	0.76	0.77
		LDM	0.51	0.39	0.47	0.47		0.59	0.59	0.60
		LGM + LDM	0.66	0.89	0.74	0.76		0.92	0.74	0.75
	MSV2	Point series	**0.95**	**0.96**	**0.99**	**1.00**	**0.98**		**0.97**	**0.97**
		LGM	0.65	0.75	0.72	0.77	0.80		0.70	0.73
		LDM	0.51	0.39	0.59	0.47	0.59		0.59	0.60
		LGM + LDM	0.77	0.95	0.78	0.83	0.84		0.77	0.73
	MTV1	Point series	**0.85**	0.88	**0.96**	**0.94**	0.62	0.59		**0.95**
		LGM	0.62	0.74	0.72	0.71	0.75	0.73		0.76
		LDM	0.50	0.50	0.50	0.50	0.50	0.50		0.50
		LGM + LDM	0.74	**0.93**	0.76	0.81	**0.80**	**0.91**		0.71
	MTV2	Point series	**0.90**	**1.00**	**0.98**	**0.93**	0.65	0.63	**0.99**	
		LGM	0.56	0.49	0.59	0.59	0.76	0.75	0.73	
		LDM	0.50	0.39	0.47	0.47	0.59	0.59	0.59	
		LGM + LDM	0.69	0.83	0.72	0.76	**0.81**	**0.86**	0.72	
Average (Point series)			**0.90**	**0.94**	**0.98**	**0.95**	**0.79**	0.76	**0.98**	**0.96**
Average (LGM)			0.56	0.64	0.69	0.69	0.74	0.73	0.72	0.64
Average (LDM)			0.50	0.50	0.47	0.50	0.52	0.53	0.50	0.51
Average (LGM + LDM)			0.81	0.79	0.79	0.83	0.78	**0.81**	0.81	0.77

representations proposed by the authors (LGM, LDM and LGM and LDM combined (LGM+LDM)) it can be seen that, in most cases, these alternatives performed badly in comparison with the point series representation. Consequently it is argued that an effective generic classifier can be produced using the proposed representation.

5 Conclusion

This paper has presented a new representation and supporting mechanism to predict feature values associated with 3D surfaces using a point series based approach. The proposed point series representation is founded on a *linearisation* of the space describing a neighbourhood surrounding a given point where the neighbourhood in turn is defined in terms of a grid. The motivation for the work was springback prediction in sheet metal forming. Two 3D surfaces (shapes) were used to evaluate the mechanism. Various forms of linearisation were considered using all the points in a linearisation or only key (corner and midway) points as well as the effect of using different grid sizes ($d = \{2.5, 5, 10, 15, 20\}$). The experiments indicated that: (i) smaller grid sizes tended to work better, (ii) the performance using 3×3, 5×5 and 7×7 neighbourhood was almost the same and (iii) that there was no significant difference in accuracy or AUC between the representations (all and key) however the key point representation offers runtime advantages. Further experiments were conducted to determine whether the linearisation could be used to produce a generic classifier, the results indicated that this was indeed the case. Excellent results were returned, 100% in terms of AUC, indicating that the point series representation is able to capture general geometric information that can successfully be employed for prediction purposes. Overall, this is a very encouraging result. For future work the intention is to conduct further experimentation with a greater variety of surfaces (shapes). The ultimate goal is to build an intelligent process model that can predict springback errors, and suggest corrections, in the context of sheet metal forming.

References

1. BS ISO 1101: 2005 Geometrical Product Specifications (GPS) Geometrical tolerancing Tolerances of form, orientation, location and run-out (2005)
2. Cafuta, G., Mole, N., Łtok, B.: An enhanced displacement adjustment method: Springback and thinning compensation. Materials and Design 40, 476–487 (2012), http://www.sciencedirect.com/science/article/pii/S026130691200252X
3. Chatti, S.: Effect of the Elasticity Formulation in Finite Strain on Springback Prediction. Computers and Structures 88(11-12), 796–805 (2010), http://www.sciencedirect.com/science/article/pii/S004579491000074X
4. Chatti, S., Hermi, N.: The Effect of Non-linear Recovery on Springback Prediction. Computers and Structures 89(13-14), 1367–1377 (2011)
5. El-Salhi, S., Coenen, F., Dixon, C., Khan, M.: Identification of correlations between 3d surfaces using data mining techniques: Predicting springback in sheet metal forming. In: Bramer, M., Petridis, M. (eds.) Research and Development in Intelligent Systems XXIX, pp. 391–404. Springer, London (2012)
6. Firat, M., Kaftanoglu, B., Eser, O.: Sheet Metal Forming Analyses With An Emphasis On the Springback Deformation. Journal of Materials Processing Technology 196(1-3), 135–148 (2008), http://www.sciencedirect.com/science/article/pii/S0924013607005353

7. Hao, W., Duncan, S.: Optimization of Tool Trajectory for Incremental Sheet Forming Using Closed Loop Control. In: 2011 IEEE Conference on Automation Science and Engineering (CASE), pp. 779–784 (2011)
8. Hilbert, D.: Ueber die stetige Abbildung einer Line auf ein Flächenstück. Mathematische Annalen 38(3), 459–460 (1891)
9. Jeswiet, J., Micari, F., Hirt, G., Bramley, A., Duflou, J., Allwood, J.: Asymmetric Single Point Incremental Forming of Sheet Metal. CIRP Annals - Manufacturing Technology 54(2), 88–114 (2005), http://www.sciencedirect.com/science/article/pii/S0007850607600213
10. Keogh, E.J., Pazzani, M.J.: Derivative dynamic time warping. In: First SIAM International Conference on Data Mining (SDM 2001) (2001)
11. Khan, M., Coenen, F., Dixon, C., El-Salhi, S.: A classification based approach for predicting springback in sheet metal forming. Journal of Theoretical and Applied Computer Science 6, 45–59 (2012)
12. Sulaiman Khan, M., Coenen, F., Dixon, C., El-Salhi, S.: Finding correlations between 3-D surfaces: A study in asymmetric incremental sheet forming. In: Perner, P. (ed.) MLDM 2012. LNCS, vol. 7376, pp. 366–379. Springer, Heidelberg (2012)
13. Liu, W., Liang, Z., Huang, T., Chen, Y., Lian, J.: Process Optimal Ccontrol of Sheet Metal Forming Springback Based on Evolutionary Strategy. In: 7th World Congress on Intelligent Control and Automation (WCICA 2008), pp. 7940–7945 (June 2008)
14. Narasimhan, N., Lovell, M.: Predicting Springback in Sheet Metal Forming: An Explicit to Implicit Sequential Solution Procedure. Finite Elements in Analysis and Design 33(1), 29–42 (1999), http://www.sciencedirect.com/science/article/pii/S0168874X99000098
15. Nasrollahi, V., Arezoo, B.: Prediction of Springback in Sheet Metal Components With Holes on the Bending Area, Using Experiments, Finite Element and Neural Networks. Materials and Design 36, 331–336 (2012), http://www.sciencedirect.com/science/article/pii/S0261306911007990
16. Peano, G.: Sur une courbe, qui remplit toute une aire plane. Mathematische Annalen 36(1), 157–160 (1890)
17. Tisza, M.: Numerical Modelling and Simulation in Sheet Metal Forming. Journal of Materials Processing Technology 151(1-3), 58–62 (2004), http://www.sciencedirect.com/science/article/pii/S0924013604003073
18. Xi, X., Keogh, E., Shelton, C., Wei, L., Ratanamahatana, C.: Fast time series classification using numerosity reduction. In: Proceedings of the 23rd International Conference on Machine Learning (ICML 2006), pp. 1033–1040. ACM, New York (2006)
19. Yoon, J., Pourboghrat, F., Chung, K., Yang, D.: Springback Prediction For Sheet Metal Forming Process Using a 3d Hybrid Membrane/Shell Method. International Journal of Mechanical Sciences 44(10), 2133–2153 (2002), http://www.sciencedirect.com/science/article/pii/S0020740302001650