Identification of Correlations Between 3D Surfaces Using Data Mining Techniques: Predicting Springback in Sheet Metal Forming

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

Abstract A classification framework for identifying correlations between 3D surfaces in the context of sheet metal forming, especially Asymmetric Incremental Sheet Forming (AISF), is described. The objective is to predict "springback", the deformation that results as a consequence of the application of a sheet metal forming processes. Central to the framework there are two proposed mechanisms to represent the geometry of 3D surfaces that are compatible with the concept of classification. The first is founded on the concept of a Local Geometry Matrix (LGM) that concisely describes the geometry surrounding a location in a 3D surface. The second, is founded on the concept of a Local Distance Measure (LDM) derived from the observation that springback is greater at locations that are away from edges and corners. The representations have been built into a classification framework directed at the prediction of springback values. The proposed framework and representations have been evaluated using two surfaces, a small and a large flat-topped pyramid, and by considering a variety of classification mechanisms and parameter settings.

1 Introduction

In sheet metal forming, especially in Asymmetric Incremental Sheet Forming (AISF), the *springback effect* is a major issue. As a result of springback the actual shape produced by the sheet metal forming process is not the same as the intended (specified) shape. The motivation for the work described in this paper is that if we can predict the springback we can apply a correction to the intended specification so as minimize the springback effect. Springback is caused by a number of factors of which the most significant is the geometry of the intended shape [1]. Further,

Subhieh El Salhi, Frans Coenen, Clare Dixon, Muhammad Sulaiman Khan
University of Liverpool, Department of Computer Science, Ashton Building, Ashton Street, Liverpool L693BX, United Kingdom
e-mail: hsselsal@liv.ac.uk, coenen@liv.ac.uk, cldixon@liv.ac.uk, mskhan@liv.ac.uk

springback is not distributed evenly over a given pshape; in practice springback is more significant over flat surfaces and with respect to some geometries than others. The solution proposed in this paper is founded on the idea of classification whereby a classifier is trained to predict springback at individual locations. Of course, for the classifier to operate correctly a suitable representation of the input data (the 3D surface of interest) is required. Two representations are proposed. The first, the Local Geometry Matrix (LGM) method, is founded on the idea of Local Binary Patterns [7], and can be used to define all possible local geometries. The second, the Local Distance Measure (LDM) method, takes into consideration the proximity of edges and corners with respect to individual points. However, before this second representation can be generated it is first necessary to identify the edges and corners of interest. Thus the contributions of this paper are as follows:

1. A mechanism for describing local 3D geometries using the concept of LGMs.
2. A mechanism for detecting edges and corners in 3D surfaces.
3. A mechanism for describing local 3D geometries using the concept of LDMs
4. A classification framework for the prediction of springback.

The rest of the paper is organized as follows. In section 2 a brief overview of related work is presented. Section 3 introduces the proposed framework for springback prediction, the section focuses on the classifier generation approach adopted. The proposed representations (LGM and LDM), in the context of the framework, are presented and discussed in Sections 3.2 and 3.3. The evaluation of the proposed framework using two surfaces, a small and a large flat topped pyramids, is presented in Section 4 in the context of a number classification paradigms and a variety of parameters. Some conclusions are then presented in Section 5.

2 Overview of related work

Asymmetric Incremental Sheet Forming (AISF) is a process for forming sheet metal parts. The potential advantage offered by processes such as AISF is a reduction in manufacturing costs and time. A comprehensive overview of AISF can be found in [9]. However, as already noted in the introduction to this paper, the main limitations of the AISF process is the springback effect. Springback can be defined as the elastic distortion that occurs as a result of the forming process so that the shape produced is not the desired shape. The springback effect is related to both manufacturing parameters and material properties [5, 15, 12]. There has been substantial reported work on springback characterization, analysis and prediction. Numerical and experimental methods have been proposed to predict springback in the context of sheet metal forming processes. The main numerical method used to analyse and predict springback for sheet metal forming is the Finite Element Method (FEM) [3, 14, 18]. Using FEM, the factors that affect the springback may be used to create a simulation model [8]. Although FEM provides a flexible simulation environment (parameters can be easily modified) FEM is an expensive and timeconsuming option [17, 5]. Furthermore, FEM is not an accurate prediction method due to the simplification assumptions that must be made [2, 3, 15]. Artificial Neural Network (ANNs) are often quoted as being a good alternative to FEM. ANNs are the most popular experimental method that has been adopted for springback prediction. Some practitioners have used the FEM model to provide the ANN with the required input data to support the training of the network [15, 11]. An alternative approach described in [13] used a genetic algorithm in order to optimize the weights for the nodes of the ANN for the purpose of springback prediction. Nevertheless, computational resources requirements remain the main limitation of ANNs [13, 6]. To the best knowledge of the authors there is no reported work on the application of classification techniques (or data mining techniques in general) for the purpose of predicting springback in the context of AISF.

3 Overview Of The Proposed Springback Prediction Framework

The input to proposed springback prediction framework is a *coordinate cloud* describing the desired 3D shape (this is typically extracted from a CAD system). The proposed framework comprises two main components: (i) data preparation and (ii) classification. During the data preparation stage the input cloud is translated into the desired representation. During the classification stage a classifier is applied to the input data to produce springback predictions. The classifier was generated in the standard manner using labelled input data. The classification stage is thus not of specific interest with respect to this paper. The novel elements of this paper are the proposed geometry representation techniques used as input to the classifier. In the context of training the desired classifier the process commences with recasting the input data (the before cloud) into an appropriate format to which further processing can be applied. A grid format is proposed for this purpose. The generation of this grid format is discussed in details in Section 3.1. As noted above classification generation requires a labelled training set, to create a labelled training set we need to compare the before cloud with the cloud produced as a result of applying the AISF (the after cloud)[1]. A description of the mechanism used to calculate the labels (error values) so as to populate a training set is presented in Section 3.1. The next stage is to represent the geometry associated with each grid point in the before coordinate grid. As already noted above, two alternative representations are considered in this paper: the LGM and the LDM methods. These are described in some detail in Sections 3.2 and 3.3, respectively.

[1] This is obtained using the GOM (Gesellschaft fr Optische Messtechnik) optical measuring tool.

3.1 Grid Representation

The proposed process, as introduced above, commences with before and after coordinate clouds (C_{in} and C_{out}). C_{in} is the point cloud for the desired shape (T), while C_{out} is the point cloud for the actual shape (T') produced as a result of application of the AISF process. A cloud point P_i is referenced in terms of a Euclidean coordinate system. Both clouds, C_{in} and C_{out}, are translated into a grid representation of the form shown in Figure 1. The size of the grid is defined in terms of a value D which represents the length of a grid square. Each grid square is referenced by the x-y coordinates of its centre point. The z value associated with each grid square is obtained by averaging the z coordinates for all the points located within it. Consequently, each grid square is represented by a central representative point described in terms of x, y, and z coordinates.

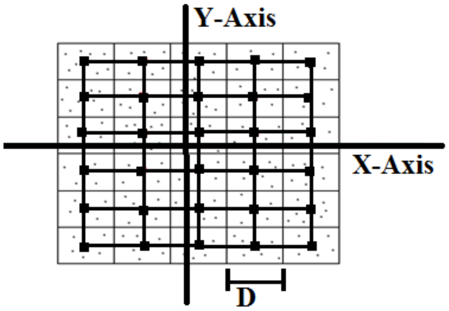

Fig. 1: The grid representation for the C_{in} and C_{out} coordinate clouds

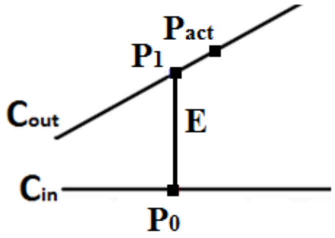

Fig. 2: Springback error calculation

The error label to be associated with each C_{in} grid point is calculated by determining the distance along the normal at each grid point from where it leaves the C_{in} shape to where it intersects the C_{out} shape as shown in Figure 2. The calculation is founded on vectors and plane geometry theory. Equation (1) shows how the distances between C_{in} and C_{out} surfaces are calculated; where P_0 and P_1 are described in terms of x, y and z coordinates, and the normal to the C_{in} plane is given by the vector (a,b,c).

$$E = \frac{|a(x_1 - x_0) + b(y_1 - y_0) + c(z_1 - z_0)|}{\sqrt{(a^2 + b^2 + c^2)}} \tag{1}$$

Thus two vectors, separated by 90°, adjoining a given centre point to two adjacent points are used to calculate a normal. Given that each point (except at corners and edges) has four neighbours (north, east, south and west) four normals can be calculated. The average of these four normals is used as the normal to a point.

Table 1: Sample training data for the small pyramid when $L = 3$ using LGM method

Z1	Z2	Z3	Z4	Z5	Z6	Z7	Z8	E
3	6	9	12	15	18	21	24	30
2	5	8	11	14	17	20	23	28
3	6	9	12	15	18	21	24	30
2	5	8	11	14	17	20	23	28
2	5	8	11	14	17	20	23	30
1	4	7	10	13	16	19	22	28

Table 2: Sample training data for the small pyramid when $L = 3$ using both the LGM and LDM methods

Z1	Z2	Z3	Z4	Z5	Z6	Z7	Z8	Dist.	E
1	4	7	10	13	16	19	22	26	28
1	4	7	10	13	16	19	22	25	28
2	5	8	11	14	17	20	23	25	28
3	5	8	12	15	18	21	24	25	29
3	6	8	12	15	18	21	24	25	28
2	5	8	11	14	17	20	23	25	28

3.2 *Local Geometry Matrix (LGM)*

The first shape description mechanism presented in this paper is the Local Geometry Matrix (LGM) method. The concept of LGMs is founded on the idea of Local Binary Patterns (LBPs). A local geometry matric is a 3×3 grid describing the locations surrounding an individual point (the point of interest is at the centre of the matrix). There are two different options for calculating the values that might be stored in an LGM (Figure 3). The first option is the difference in height (z difference) between the centre point P_0 and each of its eight neighbours P_i. The second is to store the angle, above or below the horizontal, of the lines connecting P_0 and each P_i. Whatever the case at the end of the process we have a LGM for each grid point which can be combined with an error value (calculated as described above) so that each grid point forms a record in the training set. Because we are using classifiers that operate with binary valued data we needed to discretise the training data so that each value is replaced by one of L qualitative labels used to describes the nature of the slope in each of the eight directions. An example set of qualitative labels might be $\{negative; level; positive\}$. Using this labelling, and by ordering the matrix elements (grid points) in a clockwise direction from the top left, a record might be described as follows:

$$< positive; positive; positive; level; negative; negative; negative; level; E >$$

Where E is the error value associated with the grid point that the record describes. Table 1 shows a sample set of the training data using LGM for the small pyramid when $L = 3$. The authors experiments with a number of different values for L as reported later in Section 4 of this paper.

3.3 *Local Distance Mechanism (LDM)*

The second shape description method presented in this paper is the Local Distance Measure (LDM) method. This is founded on the observation that the springback tends to be greater further from edges. The idea is therefore to describe each grid

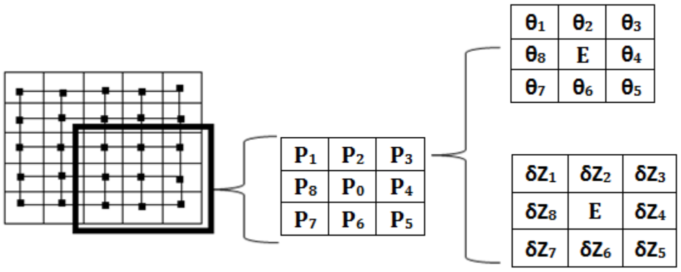

Fig. 3: LGM Representation

square centre in terms of its distance from its nearest edge. This first necessitates identifying the edges in the input grid. Edge detection was conducted by first dividing C_{in} into K regions using the well know K-means algorithm ($K = 4$ was empirically chosen and used with respect to the evaluation described later in this paper). The average angular value for the normals in each region was first determined. The angular difference between the normal at each centre point and the average for the region in which the point was located was then calculated. If the angular difference was greater than some tolerance measure, σ, the point was considered to be located on an edge. The process is illustrated in Figure 4. Once the edges have been detected, the minimum distance between each grid point and its closest edge was determined simply by adopting a "region growing" process. The result is a set of records each describing a grid square location in terms of its edge distance and its associated error label (springback). Clearly, there are two main factors that affect the process of edge detection:

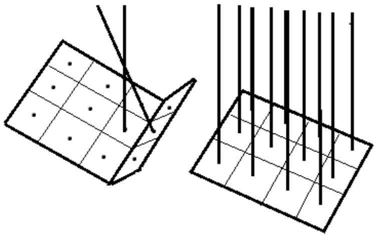

Fig. 4: Edge detection

Table 3: Sample training data for the small pyramid when $L = 3$ using LDM method

Distance	Error
24	30
23	28
24	30
23	28
24	28
23	29
22	30
22	28

- **The tolerance value σ**: As σ is increased, the number of points identified as edge points decreases. Conversely as σ is decreased, the number of points identified as edge points increases. This is illustrated in Figures 5 and 6 where it can be seen that more points are identified as edge points when σ=15 than when σ=5. Therefore, σ should be carefully chosen.

- **The grid size** (*D*): The greater the grid size the more difficult it is to detect edges. For a relatively large value of *D*, the grid square will cover a larger area. Therefore the normals are more likely to be parallel than if a small value of *D* is used. This is illustrated in Figures 7 and 8 where σ has been kept constant (σ=9) and *D* set at 5 and 15 respectively. In Figure 7 the edges are easily detected as there is an obvious variation in the angular differences. However, in Figure 8 all the points are identified as edges as all the normals are different (by at least a σ). Table 3 introduces a sample of the training data set for the small pyramid using LDM where each value for each attribute is replaced by one of the qualitative labels for $L = 3$. Table 2 presents another example for the training data set, for the same small pyramid, using the same number of labels ($L = 3$) where LDM and LGM methods are used together.

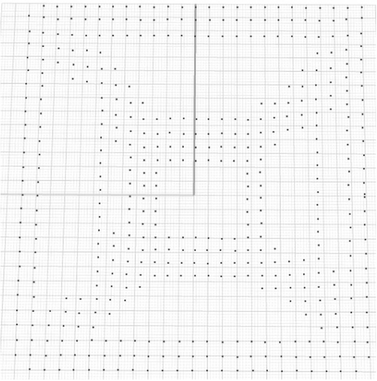

Fig. 5: Large pyramid with D=5 and σ=5

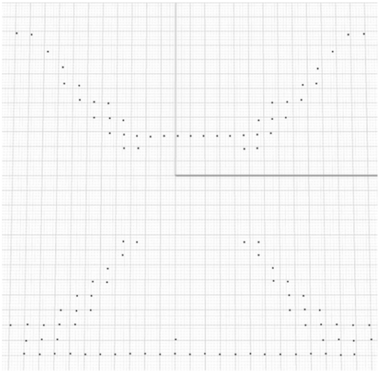

Fig. 6: Large pyramid with D=5 and σ=15

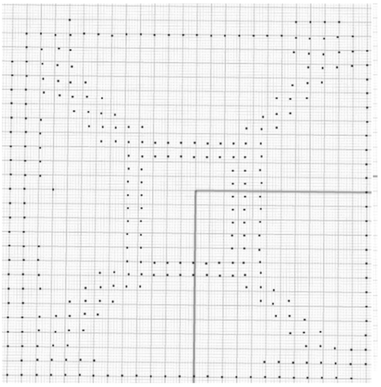

Fig. 7: Small pyramid with D=5 and σ=9

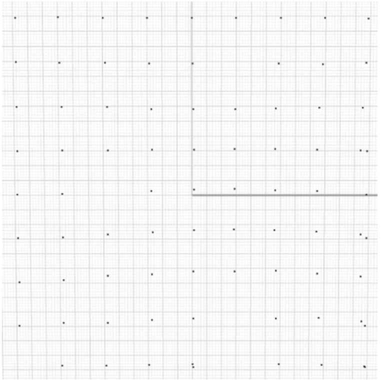

Fig. 8: Small pyramid with D=15 and σ=9

4 Evaluation

To evaluate the proposed springback prediction framework two sample surfaces were used, a small and a large flat topped pyramid. The small pyramid base size is 140 mm whereas the base size for the large pyramid is 350 mm. Both were defined in terms of before coordinate clouds extracted from a CAD system. These shapes were then manufactured [2] in steel using AISF. After clouds were then generated from the manufactured parts. These before and after clouds were then processed as described above to generate a number of data sets to be used for evaluation purposes. Broadly these data sets can be divided into three categories according to the representation used: (i) LGM, (ii) LDM and (iii) a combination of LGM and LDM. Six sets of experiments were conducted as follows. The first two sets of experiments, reported in Section 4.1 and 4.2, were intended to determine the performance of the proposed approach when applied individually to the small and large pyramids. The next two sets of experiments, reported in Sections 4.3 and 4.4, were designed to determine whether a classifier trained on the small pyramid could be applied to the large pyramid and vice-versa. The significance of these last two experiments was that it would serve to go some way to establishing whether we could produce a generic classifier trained from some appropriate shape that could be applied to other shapes. Some statistics concerning the data sets are presented in Table 4. The results from these four experiments are reported in the following four subsections. Sub-sections 4.5 and 4.6 then analyse the effect of using different values for D (grid size) and L (number of qualitative labels). For the experiments three different classifier generators were considered: (i) Bayes [10], (ii) JRIP [4] and C4.5 [16]. A range of values of L of 3, 5 and 7 was used; and a range of values for D of 5, 10, 15 and 20. For each experiment the results were recorded in terms of percentage accuracy.

Table 4: Number of records generated for the large and small pyramids using different values for D

D	Small Pyramid	Large Pyramid
5	783	4901
10	196	1223
15	78	528
20	46	289

[2] The small and the large pyramid were generated purely for research purpose as it is a commonly used shape with respect to AISF related research.

4.1 Small pyramid

Table 5 presents the results obtained when applying the proposed techniques to the small pyramid test shape. The reported accuracy values are average values generated using Ten Cross Validation (TCV). From the table we can observe that:

- The highest accuracy obtained was 87.01%, using $D = 15$ and $L = 3$.
- The best performing techniques were the LGM and the combined LGM and LDM technique; the LDM technique on its own did not perform as well as the other two techniques.
- There was no significant distinction between the operation of the three of classification algorithms considered, especially in the context of the LDM technique; however an argument can be made that Bayes and JRIP outperformed C4.5.

Table 5: Accuracy results for the small pyramid

Method	L	D=5			D=10			D=15			D=20		
		Bayes	JRIP	C4.5	Bayes	JRIP	C4.5	Bayes	JRIP	C4.5	Bayes	JRIP	C4.5
LGM	3	61.13	84.02	84.02	71.28	85.13	86.67	**87.01**	**87.01**	85.71	57.78	75.56	75.56
	5	57.80	77.88	77.49	62.05	77.95	77.44	55.84	71.43	72.73	55.56	68.89	51.11
	7	66.88	75.58	75.96	67.18	72.82	68.72	58.44	67.53	76.62	51.11	51.11	44.44
LDM	3	84.53	84.53	84.53	84.11	84.10	84.10	75.33	75.32	75.32	75.56	75.56	75.56
	5	71.87	71.87	71.87	73.85	73.85	73.85	61.04	61.04	61.03	66.67	66.67	66.67
	7	67.65	67.65	67.65	69.23	69.23	69.23	53.25	49.35	49.35	60.00	60.00	57.78
LGM and LDM	3	61.13	84.27	84.02	74.36	85.13	86.67	**87.01**	**87.01**	85.71	57.78	73.33	71.11
	5	58.06	77.49	77.88	62.05	77.44	76.92	57.14	71.43	72.73	57.78	66.67	51.11
	7	66.90	75.32	75.70	67.67	72.82	69.74	59.74	62.34	76.62	55.56	53.33	44.44

From the table we can also observe that the accuracy associated with the LGM and the combination techniques increased gradually as D was increased from 5 to 15, then dropped of when $D = 20$. It was conjectured that this was because when the grid was too small the geometry could not be well defined and when it got too large the geometry was too coarsely defined. It is also interesting to note that the LDM method is more stable in that it features smaller fluctuations in accuracy as D is increased from 5 to 20.

4.2 Large pyramid

Table 6 presents the results obtained when the proposed techniques were applied to the large pyramid. Again the results were generated using TCV. From the table it can be seen that:

- The best overall accuracy result obtained was 99.16%, using $D = 5$ and $L = 3$. This is an extremely high accuracy.
- Marginally better results were obtained using LGM and the combination technique, although there was not a significant distinction between the techniques.
- Out of the three classification algorithms considered the best result was obtained using C4.5, but the margin between different data classification techniques was very small, hence it can be argued that there was no significant difference between the classifiers.

Overall the experiments on both the small and large pyramids indicated that very good error predictions could be obtained from application of the proposed framework, especially in the case of the large pyramid (which included more points for the same value of D as indicated in Table 4).

Table 6: Accuracy results for the large pyramid

Method	L	D=5			D=10			D=15			D=20		
		Bayes	JRIP	C4.5	Bayes	JRIP	C4.5	Bayes	JRIP	C4.5	Bayes	JRIP	C4.5
LGM	3	95.12	99.10	**99.16**	91.65	97.95	98.04	89.94	98.29	98.29	88.19	97.22	97.22
	5	89.27	99.10	99.10	86.82	96.48	96.64	86.91	97.15	97.34	86.11	96.88	96.88
	7	86.67	98.69	98.67	86.50	95.01	95.34	85.77	95.83	96.02	81.60	95.49	95.83
LDM	3	98.94	98.94	98.94	97.71	97.71	97.71	98.29	98.29	98.29	97.22	97.22	97.22
	5	98.78	98.78	98.78	96.40	96.40	96.40	96.77	96.77	96.77	96.88	96.88	96.88
	7	98.69	98.69	98.69	95.09	95.09	95.09	96.02	96.02	96.02	95.49	95.49	95.49
LGM and LDM	3	95.04	99.10	**99.16**	91.65	97.95	98.04	90.13	98.29	98.29	88.19	97.22	97.22
	5	89.27	99.10	99.10	86.82	96.48	96.64	87.10	97.15	97.34	86.11	96.88	96.88
	7	86.67	98.69	98.67	86.58	94.84	95.50	85.77	95.83	96.02	81.60	95.49	95.83

4.3 Training on the small pyramid and testing on the large pyramid

The results obtained when building a classifier using the small pyramid and applying it on the large pyramid are presented in Table 7. From the table it can be seen that:

- The best accuracy obtained was 98.94%, using $D = 5$ and $L = 3$. This is again an excellent result.
- The number of labels (3, 5, 7) did not appear to play a significant role although L=3 produced best results (but only with a small margin). However, it should be noted that if we wish to apply corrections to the input cloud we probably need to use a high value for L.
- The LDM and the combined technique produced better results than the LGM technique.
- Out of the three classification algorithms considered there was no obvious distinction between their operations.

Table 7: Accuracy results when training on the small and testing on the large pyramid

Method	L	D=5			D=10			D=15			D=20		
		Bayes	JRIP	C4.5	Bayes	JRIP	C4.5	Bayes	JRIP	C4.5	Bayes	JRIP	C4.5
LGM	3	94.90	98.65	98.65	94.60	97.71	97.71	92.22	93.36	93.36	96.18	97.22	88.68
	5	93.12	95.96	95.94	93.93	93.94	93.94	89.56	93.93	92.22	90.63	96.88	89.58
	7	94.39	96.16	96.00	90.00	90.92	93.70	95.69	94.50	91.84	89.93	95.49	94.10
LDM	3	**98.94**	**98.94**	**98.94**	97.71	97.71	97.71	98.29	98.30	98.30	97.22	97.22	97.22
	5	98.78	98.78	98.78	96.40	96.40	96.40	96.77	96.77	96.78	96.88	95.83	95.83
	7	98.69	98.69	98.69	95.09	95.09	95.09	96.02	96.02	96.02	95.48	95.49	95.49
LGM and LDM	3	**98.94**	98.65	98.80	94.60	97.71	97.71	92.22	93.17	93.36	94.94	98.65	92.36
	5	93.12	95.92	95.94	94.01	93.94	94.35	90.32	93.93	92.22	91.67	95.83	89.58
	7	94.39	96.16	96.04	91.57	91.00	93.70	95.88	93.36	91.84	92.71	95.49	94.10

4.4 Training on the large pyramid and testing on the small pyramid

Table 8 shows the results produced when building a classifier using the large pyramid and applying it to the small pyramid. From the table it can be observed that:

- The best accuracy obtained is 84.53%, again using $D = 5$ and $L = 3$.
- There was no significant distinction between the operation of the LGM, LDM and combination techniques, although LDM did produce marginally better results.
- Out of the three data mining techniques considered, there was no significant performance distinction between them.

The results obtained from this experiment, and the previous experiment, indicated that it might be possible to build a generic classifier. It is conjectured that both the small pyramid and the large pyramid contain sufficient examples of all different possible geometries to allow for effective classification, with the larger pyramid producing slightly better performing classifiers than those produced using the small pyramid.

Table 8: Accuracy results when training on the large and testing on the small pyramid

Method	L	D=5			D=10			D=15			D=20		
		Bayes	JRIP	C4.5	Bayes	JRIP	C4.5	Bayes	JRIP	C4.5	Bayes	JRIP	C4.5
LGM	3	59.08	84.02	84.02	29.74	84.10	84.10	36.36	55.84	55.84	46.67	75.56	75.56
	5	41.18	71.61	71.36	16.92	73.85	73.85	33.77	53.25	53.25	31.11	68.89	68.89
	7	33.12	67.65	57.54	13.85	69.23	68.21	22.08	53.25	51.95	22.22	53.33	60.00
LDM	3	**84.53**	**84.53**	**84.53**	84.10	84.10	84.10	75.32	75.32	75.32	75.56	75.56	75.56
	5	71.87	71.87	71.87	73.85	73.85	73.85	61.04	61.04	61.04	68.89	68.89	68.89
	7	67.65	67.65	67.65	69.23	69.23	69.23	53.25	53.25	53.25	60.00	60.00	60.00
LGM and LDM	3	59.08	84.02	84.02	29.74	84.10	84.10	37.67	55.84	55.84	46.67	75.56	75.56
	5	41.18	71.61	71.36	17.44	73.85	73.85	32.47	53.25	53.25	31.11	68.89	68.89
	7	33.12	67.65	57.54	13.85	69.23	68.21	22.08	53.25	51.95	22.22	53.33	60.00

4.5 The effect of grid size (D)

Clearly, from the foregoing, the grid size parameter, D, has an effect. In this subsection a brief analysis of the effect that the value for D has on classification performance is presented. Table 9 shows a comparison between the percentage accuracy and D values for the best performing combinations from the foregoing experiments when $L = 3$. These are identified using the letters A, B, C and D as follows:

A. Training and testing on the large pyramid using, C4.5 and LGM.
B. Training and testing on the small pyramid using JRIP and LGM.
C. Training on the large and testing on the small using JRIP and LDM.
D. Training on the small and testing on the large using Bayes and LDM.

Table 9: The effect of D when $L = 3$

	D=5	D=10	D=15	D=20
A	99.16	98.04	98.29	97.22
B	84.02	85.13	87.01	75.56
C	84.53	84.10	75.33	75.56
D	98.94	97.71	98.29	97.22

From Table 9 it can be observed that when training on the large pyramid (which produced the better performance) accuracy dropped as D was increased. In the case where we trained and tested on the small pyramid (case B) accuracy peaked at $D = 15$, while in case C accuracy peaked at $D = 5$. Overall we can conclude that $D = 5$ tends to produce a better result.

4.6 The effect of label set size (L)

This section completes the evaluation with a brief analysis of the effect of the label set size (L). The analysis was conducted by considering the best performing combinations when $D = 5$ [3]; these are again identified using the letters A, B, C and D as follows:

A. Training and testing on the large pyramid using C4.5 and LGM.
B. Training and testing on the small pyramid using C4.5 and LDM.
C. Training on the large and testing on the small using JRIP and LDM.
D. Training on the small and testing on the large using Bayes and LDM.

[3] It is acknowledged that the best result for the small pyramid was obtained when $D = 15$, however we wished to conduct the analysis concerning L by maintaining D at a constant value.

The results of the analysis are presented in Table 10. In this case we can note that the value of L has little effect with respect to case A and D, while in cases B and C accuracy decreases as the number of labels increases.

Table 10: The effect of L when $D = 5$

	L=3	L=5	L=7
A	99.16	99.10	98.67
B	84.53	71.87	67.65
C	84.53	71.87	67.65
D	98.94	98.78	98.69

5 Conclusion

This paper has presented a new framework to identify the correlations between 3D surfaces using two different mechanisms for shape description, the first founded on the concept of Local Geometry Matrixes (LGMs) and the second on Local Distance Measures (LDMs). Both mechanisms, and a combination of the two, were evaluated by applying them to two manufactured surfaces, a small and a large flat-topped pyramid. The main finding of the work is that good (very good in some cases) classification accuracy results can be produced not only when the classifier is trained and tested on an identical shape but also when the classifier is trained on one shape and tested on another where every possible pattern that can exist using LGM method is included in the classifier when the pyramid shape considered. The significance of the latter is that this is an indication (further experimentation is clearly required) that we can build a classifier that encompasses all possible geometries by considering a suitable shape, which can then be generally applied. This has implications if we want to build a system that can suggest corrections to be applied to before clouds that can serve to limit the effect of springback. Other significant findings included: (i) confirmation that small grid sizes produce a better performance than large grid sizes, (ii) that both the LGM and LDM techniques worked well and that it could not be argued that there was a significant difference in their operation, and (iii) that the choice of classification algorithm did not make a significant impact. Overall this is a very encouraging result. For future work a new surface representation approach founded on the concept of time series is currently under investigation. The intention is also to conduct further experimentation with a greater variety of surfaces (shapes) and a detailed comparison between the LGM, LDM, the combination of both and the new time series surface representation methods. The ultimate goal is to build an intelligent process model that can predict springback errors and suggest corrections to before coordinate clouds.

Acknowledgements This work was partially supported by the EU project Innovative Manufacturing of complex Ti Sheet Components (INMA), grant agreement number 266208.

References

1. G. Cafuta, N. Mole, and B. Łtok. An enhanced displacement adjustment method: Springback and thinning compensation. *Materials and Design*, 40:476 – 487, 2012.
2. S. Chatti. Effect of the Elasticity Formulation in Finite Strain on Springback Prediction. *Computers and Structures*, 88(1112):796 – 805, 2010.
3. S. Chatti and N. Hermi. The Effect of Non-linear Recovery on Springback Prediction. *Computers and Structures*, 89(13-14):1367 – 1377, 2011.
4. W. Cohen. Fast effective rule induction. In *Twelfth International Conference on Machine Learning*, pages 115–123. Morgan Kaufmann, 1995.
5. M. Firat, B. Kaftanoglu, and O. Eser. Sheet Metal Forming Analyses With An Emphasis On the Springback Deformation. *Journal of Materials Processing Technology*, 196(1-3):135 – 148, 2008.
6. Z. Fu, J. Mo, L. Chen, and W. Chen. Using Genetic Algorithm-Back Propagation Neural Network Prediction and Finite-Element Model Simulation to Optimize the Process of Multiple-Step Incremental Air-Bending Forming of Sheet Metal. *Materials and Design*, 31(1):267 – 277, 2010.
7. Z. Guo, L. Zhang, and D. Zhang. A Completed Modeling of Local Binary Pattern Operator for Texture Classification. *IEEE Transactions on Image Processing*, 19(6):1657–1663, 2010.
8. W. Hao and S. Duncan. Optimization of Tool Trajectory for Incremental Sheet Forming Using Closed Loop Control. In *Automation Science and Engineering (CASE), 2011 IEEE Conference on*, pages 779 –784, 2011.
9. J. Jeswiet, F. Micari, G. Hirt, A. Bramley, J. Duflou, and J. Allwood. Asymmetric Single Point Incremental Forming of Sheet Metal. *CIRP Annals - Manufacturing Technology*, 54(2):88 – 114, 2005.
10. G. John and P. Langley. Estimating Continuous Distributions in Bayesian Classifiers. In *Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 338–345, San Mateo, 1995. Morgan Kaufmann.
11. R. Kazan, M. Firat, and A. Egrisogut Tiryaki. Prediction of Springback in Wipe-Bending Process of Sheet Metal Using Neural Network. *Materials and Design*, 30(2):418 – 423, 2009.
12. W. Liu, Z. Liang, T. Huang, Y. Chen, and J. Lian. Process Optimal Ccontrol of Sheet Metal Forming Springback Based on Evolutionary Strategy. In *Intelligent Control and Automation, 2008. WCICA 2008. 7th World Congress on*, pages 7940 –7945, June 2008.
13. W. Liu, Q. Liu, F. Ruan, Z. Liang, and H. Qiu. Springback Prediction for Sheet Metal Forming Based on ga-ann Technology. *Journal of Materials Processing Technology*, 187-188:227 – 231, 2007.
14. N. Narasimhan and M. Lovell. Predicting Springback in Sheet Metal Forming: An Explicit to Implicit Sequential Solution Procedure. *Finite Elements in Analysis and Design*, 33(1):29 – 42, 1999.
15. V. Nasrollahi and B. Arezoo. 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.
16. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA, 1993.
17. M. Tisza. Numerical Modelling and Simulation in Sheet Metal Forming. *Journal of Materials Processing Technology*, 151(1-3):58 – 62, 2004.
18. J. Yoon, F. Pourboghrat, K. Chung, and D. Yang. Springback Prediction For Sheet Metal Forming Process Using a 3d Hybrid Membrane/Shell Method. *International Journal of Mechanical Sciences*, 44(10):2133 – 2153, 2002.