SoftwareX 27 (2024) 101837

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

HICA: A MATLAB-based hierarchical image clustering algorithm for classifying parts suitable for additive and traditional manufacturing technologies

Baris Ördek, Yuri Borgianni *

Faculty of Engineering, Free University of Bozen-Bolzano, Piazza Università, 1, 39100 Bolzano, Italy

ARTICLE INFO

Keywords:
MATLAB
Hierarchical clustering
Image processing
Additive manufacturing
Process selection
SolidWorks
CAD

ABSTRACT

This paper presents an image clustering algorithm that classifies parts to be fabricated using traditional and additive manufacturing (AM) technologies. The proposed algorithm is a MATLAB-based software tool that clusters 3D CAD models of parts considering their geometry only. The algorithm can classify image datasets, CAD datasets, and combined datasets that contain both images and CAD models. The software tool reduces the time and effort spent during process selection by offering a preselected set of parts that are more suitable for AM. The software tool is aimed at supporting decision-making for traditional manufacturing companies that consider expanding their production capability by introducing AM processes in their production facilities. The HICA software tool expands the scope of scientific applications in manufacturing process selection by providing an unsupervised approach that does not require data labelling. The tool is made available as a MATLAB function through a permanent link.

Metadata

Nr	Code metadata description	
C1	Current code version	v3.0
C2	Permanent link to code/ repository used for this code version	https://github.com/brsordk/HICA
C3	Permanent link to reproducible capsule	NA
C4	Legal code license	GNU General Public License v3.0
C5	Code versioning system used	NA
C6	Software code languages, tools and services used	MATLAB & Any CAD processing software
C7	Compilation requirements, operating environments and dependencies	MATLAB 2017a and later releases, Statistics and Machine Learning Toolbox, Computer Vision Toolbox & Image Processing Toolbox. Any CAD software that extract images automatically.
C8	If available, link to developer documentation/manual	All information regarding the use and application are given in GitHub.
C9	Support email for questions	boerdek@unibz.it / barisordek@gmail.com

1. Motivation and significance

Integrating advanced manufacturing technologies in the industry, specifically additive manufacturing (AM), gives the opportunity to avail of fabrication techniques compatible with complex shapes. The design freedom enabled by AM largely surpasses the one exhibited by traditional manufacturing (TM) technologies [1,2], for which constraints are rather typically focused on. Otherwise said, detailed or very complex shapes that were previously unachievable or costly to manufacture using traditional techniques can now be easily fabricated using AM [3]. This opens new research avenues not only for engineers and designers but also for companies seeking to expand their manufacturing capabilities by introducing AM processes in the set of used fabrication processes. Integrating AM into a company's production asset creates a decision-making problem that can be tackled with manufacturing process selection (MPS) methods. This problem might extend to established products that have been traditionally fabricated through TM, challenging rooted practices.

MPS is one of the critical decisions to be made during product

* Corresponding author.
E-mail address: yuri.borgianni@unibz.it (Y. Borgianni).

https://doi.org/10.1016/j.softx.2024.101837
Received 4 December 2023; Received in revised form 13 June 2024; Accepted 23 July 2024
Available online 27 July 2024

development [4] because it is necessary to contemplate technical, economic, and environmental dimensions of the manufacturing process [5]. MPS is typically made by experts [6]. However, intelligent systems are increasingly considered as a viable option, especially where human expertise is limited [4]. The use of intelligent systems in MPS is nevertheless limited by the heavy reliance on engineering drawings and tacit human knowledge [7]. The available software applications are relatively constrained since they rely on rule-based systems [8]. Machine Learning (ML) is commonly used in manufacturing industries to minimize human supervision and lead time of a product [9]. These facts have encouraged the development of systems for part classification and process selection using ML algorithms, which are typically based on CAD models and images.

Table 1 shows approaches that have implemented different ML algorithms and programming languages with the objective of MPS or sorting parts. Notably, almost all tools were generated using Python, and some applications were developed using C++ (see Table 1). ML-based systems are trained using labelled data (supervised learning (SL)) and unlabelled data (unsupervised learning (UL)). Most research contributions have exploited SL algorithms to classify images or CAD models. Even though SL algorithms are more successful and preferred in the literature, they have to be applied to labelled datasets in which it is necessary to label each data point carefully.

Data labelling is a significant process that involves assigning information tags or labels to raw data (i.e., unlabelled data). The availability and accessibility of raw datasets are increasing with the rapid advancement of the industry. In practice, raw datasets are usually labelled by human experts in the field; however, data labelling is time-consuming and expensive [15]. This sheds light on the convenience and the necessity of using UL to generate simple classifications. Hence, the HICA algorithm presented in this paper uses unlabelled datasets to cluster objects based on their geometry. It aims to advance the literature on intelligent systems supporting MPS. This can take place as the algorithm offers a preselection of parts more suitable for AM processes, which makes the algorithm time-efficient and cost-effective.

2. Software description

2.1. Software architecture

The HICA algorithm is built using CAD and MATLAB [16] software. The overall methodology behind the development of HICA is presented in Fig. 2. The algorithm uses images and CAD models as inputs. Images can be directly processed by the HICA algorithm while, in the case of using CAD models as inputs, a CAD processing software is required to convert CAD models into corresponding images. It is highly recommended to select a CAD processing software that can automatically convert CAD models into images without backgrounds and shadows.

In the application presented in this paper, an isometric view image of each CAD model was extracted to have a consistent image dataset. This view is a standard representation of many CAD software tools aimed to show geometric characteristic of parts in all three directions. The data presented in Section 3 was prepared using SolidWorks Task Scheduler, which extracts images from CAD models automatically. Most diffused CAD software applications have access to CAD kernel systems that can automatically generate the necessary snapshots from CAD models (e.g., NXOpen from Siemens NX, Catia V5, Autodesk Fusion).

In the MATLAB environment, the algorithm is structured into five steps. First, part images are stored as an image database with file names stored as labels. The second step is preprocessing the images, where each image is converted from RGB colour to grayscale and resized to have the same characteristics and avoid misclassifications. The third step includes feature extraction, which can be expressed as an alternative version of an image that highlights essential portions. In this research, the external geometry of the parts was highlighted and extracted. The histogram of oriented gradients (HOG) feature extraction methodology was employed to extract these features since this method proved to be successful in the literature for object detection [17]. Fig. 1 shows an example of the HOG visualization of a part from the input dataset. The external geometry of each part was highlighted using the HOG visualization as shown in Fig. 1.

In the fourth step, the extracted features are stored as a matrix, where each matrix column represents an image. Eventually, the cosine distance between each column of this matrix is calculated using Eq. (1).

$$\text{Cosine Distance} = \frac{x \cdot y}{\| x \| \cdot \| y \|} \tag{1}$$

where x and y are 2 column vectors from the feature matrix.

The resulting distances were then used to separate the clusters using the complete linkage method, which depends on the maximum distances between clusters. This linkage method was selected because it can generate spherical compact clusters and is more resilient to fluctuations and noise [18].

Hierarchical clustering was preferred in the development of the HICA alongside HOG because of the following reasons:

- HOG features capture local gradient information on an image which is beneficial for hierarchical clustering since it can identify similar structures or patterns in images [3].
- HOG provides a compact representation of an image by converting local gradients into a feature vector. This dimensionality reduction can benefit the hierarchical clustering positively [19].

Moreover, hierarchical clustering is preferred over other clustering methods, such as DBSCAN and HDBSCAN, for the following reasons:

- Hierarchical clustering does not require specification parameters, while DBSCAN and HDBSCAN need careful selection of some parameters that may require expertise [20,21].
- Hierarchical clustering produces a dendrogram that allows the user to understand the hierarchical structure of the data [20,21].
- Hierarchical clustering can be applied to a variety of data types, including binary, continuous, and categorical data, depending on the preferred distance metric (e.g., HICA uses cosine distance) [20,21].

The final step of the algorithm is to visualize and cut the dendrogram to form the cluster numbers initially specified by the user. The HICA algorithm can cluster the parts up to five clusters, which was initially selected randomly. However, the algorithm can be modified according to specific needs.

2.2. Software functionalities

The HICA algorithm requires eight MATLAB functions to be able to cluster parts. This section describes the application of these functions to develop the HICA algorithm.

Table 1
Tools proposed in the literature to classify images and CAD models.

Author	Year	Language	ML Algorithm	Dataset	Reference
Li et al.	2023	Python	Graph Neural Network	3D CAD	[10]
Dai et al.	2022	Python	Self-supervised Pairing Image Clustering	Image	[11]
Miles et al.	2022	Python	Recursive Neural Network	3D CAD	[12]
Qin et al.	2014	Python	Deep Neural Network	3D CAD	[13]
Hou et al.	2005	C++	Support Vector Machines	3D CAD	[14]

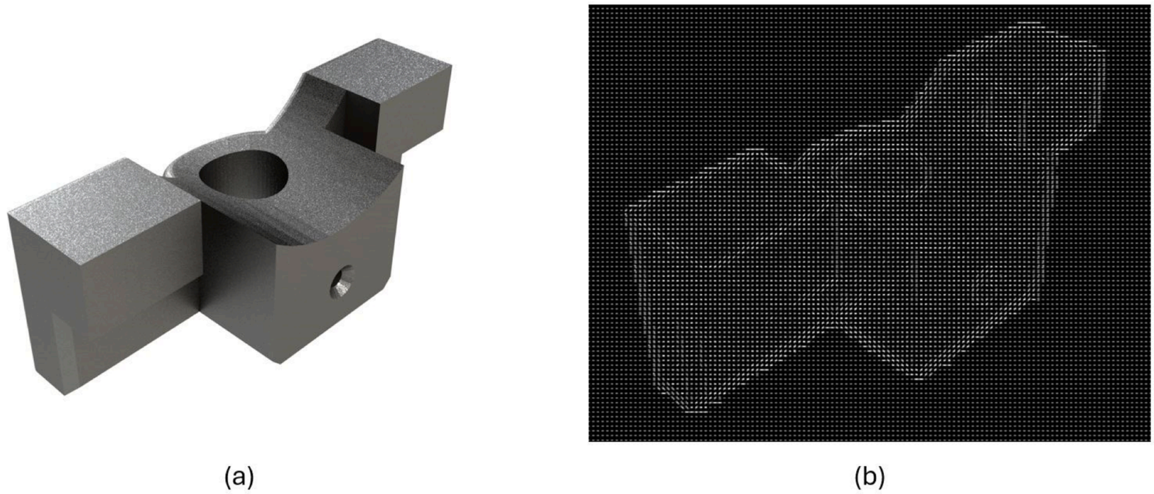


Fig. 1. (a) an example part from the input dataset and (b) the HOG visualization of the example part.

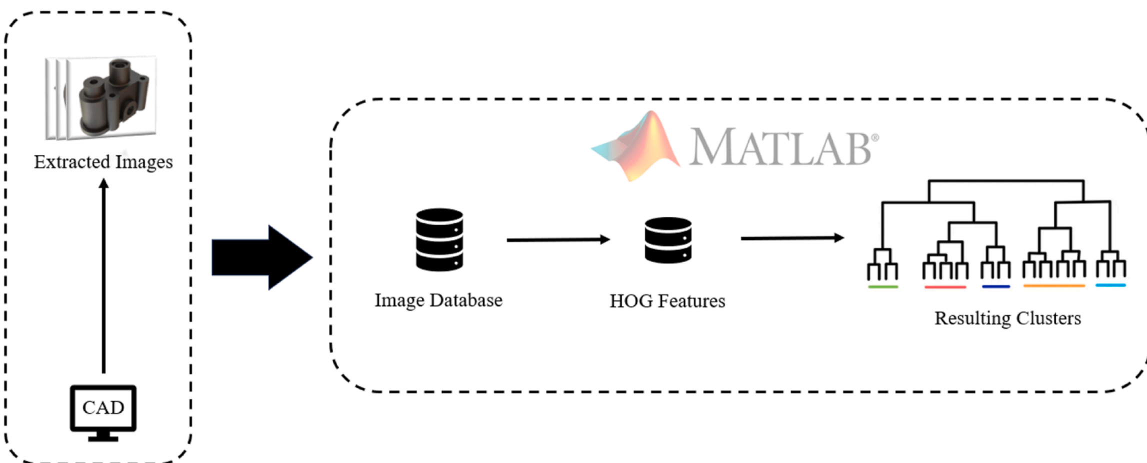


Fig. 2. The overview of the HICA software architecture.

- *imageDatastore:* This function is used to create an image database. Input images are used to create a database in the HICA algorithm using this function.
- *imread:* This function is used to read information from an image. Every image is inserted into the HICA algorithm using this function.
- *rgb2gray:* this function converts a colourful image into grayscale image. Every image in the database is converted into grayscale using this function.
- *extractHOGFeatures:* This function returns a feature vector with hog feature information. It also allows the user to visualize the hog features. The HOG features are extracted using this function in the HICA algorithm.
- *pdist:* It is a function that calculates the pairwise distance between observations; 'cosine' distance is calculated in the HICA algorithm using this function.
- *linkage:* This function creates a hierarchical tree considering a specific method. The HICA algorithm uses the 'complete' linkage method.
- *dendrogram:* This function generates the dendrogram figure using the linkage method specified in 'linkage' function.
- *writetable:* This function is used to export a table as a "text" or "MS Excel" file. The HICA algorithm exports the table as an MS Excel file with ".xlsx" extension.

3. Illustrative example

The HICA algorithm is used in the same way for all MATLAB functions by inserting the necessary input arguments, as explained at the beginning of the script (available in the GitHub repository). The HICA algorithm is tested using a dataset that contains 80 parts, for which indications about used manufacturing technology was declared or could be inferred through available information (37.5 % AM and 62.5 % TM). These parts were taken from freely accessible online databases (e.g., GrabCAD). They were subsequently assigned file names according to the manufacturing process used for their fabrication, AM or TM.

After specifying the location of the folder containing the files and the number of desired clusters, the HICA algorithm was started by using the run function of MATLAB.

The results of the HICA algorithm can be viewed as a dendrogram and two pictures, one for each cluster including the images of the processed models. Based on the 80 parts selected for the application of HICA, Fig. 3 shows the dendrogram that includes the two resulting clusters in light blue and red coloured lines for Cluster 1 (parts more suitable to be fabricated using TM) and Cluster 2 (parts more suitable for fabrication with AM), respectively. The designation of the two clusters as more representative for TM and AM parts has to be made manually, with the possible support of the generated pictures. In Fig. 3, the eighty input parts are ordered in the abscissas, while the ordinate shows the distance between the clusters.

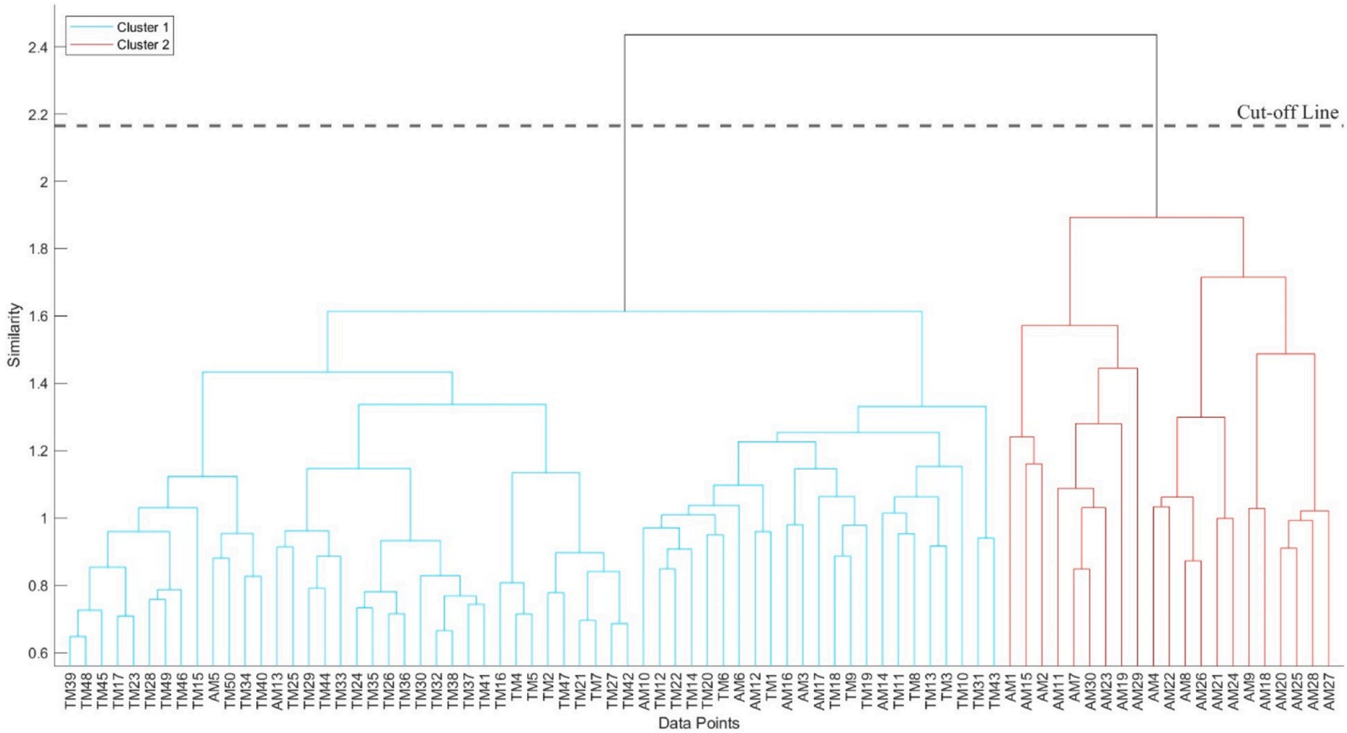


Fig. 3. The HICA results shown as a two-cluster dendrogram.

The resulting clusters are compared with the actual manufacturing processes, as shown in Fig. 4. The performance of the HICA algorithm was calculated as 88.75 %, considering the 80 input parts. The misclassified parts are highlighted with red coloured cells in Fig. 4. All of them involve parts for which AM was considered as more suitable and wrongly assigned to the TM cluster.

4. Impact

As evidenced in Section 1, several articles suggest a solution to MPS by considering supervised ML algorithms. However, SL requires labelled datasets which are difficult to obtain, and data labelling is expensive. Thus, the HICA algorithm is designed to cluster unlabelled datasets, and it is one of the first MATLAB-based UL applications used to differentiate parts more suitable for AM and TM. This unsupervised approach and the ability to differentiate parts by only considering part geometry are the novelty of the proposed algorithm.

The primary target is represented by the traditional (or conventional) manufacturing companies with large production inventories that intend to reorient their production capabilities by benefitting from AM processes. The HICA algorithm can be effectively used to preselect a set of parts that are more suitable to be fabricated using AM technologies. Hence, the algorithm offers a quick and cost-effective alternative for part identification.

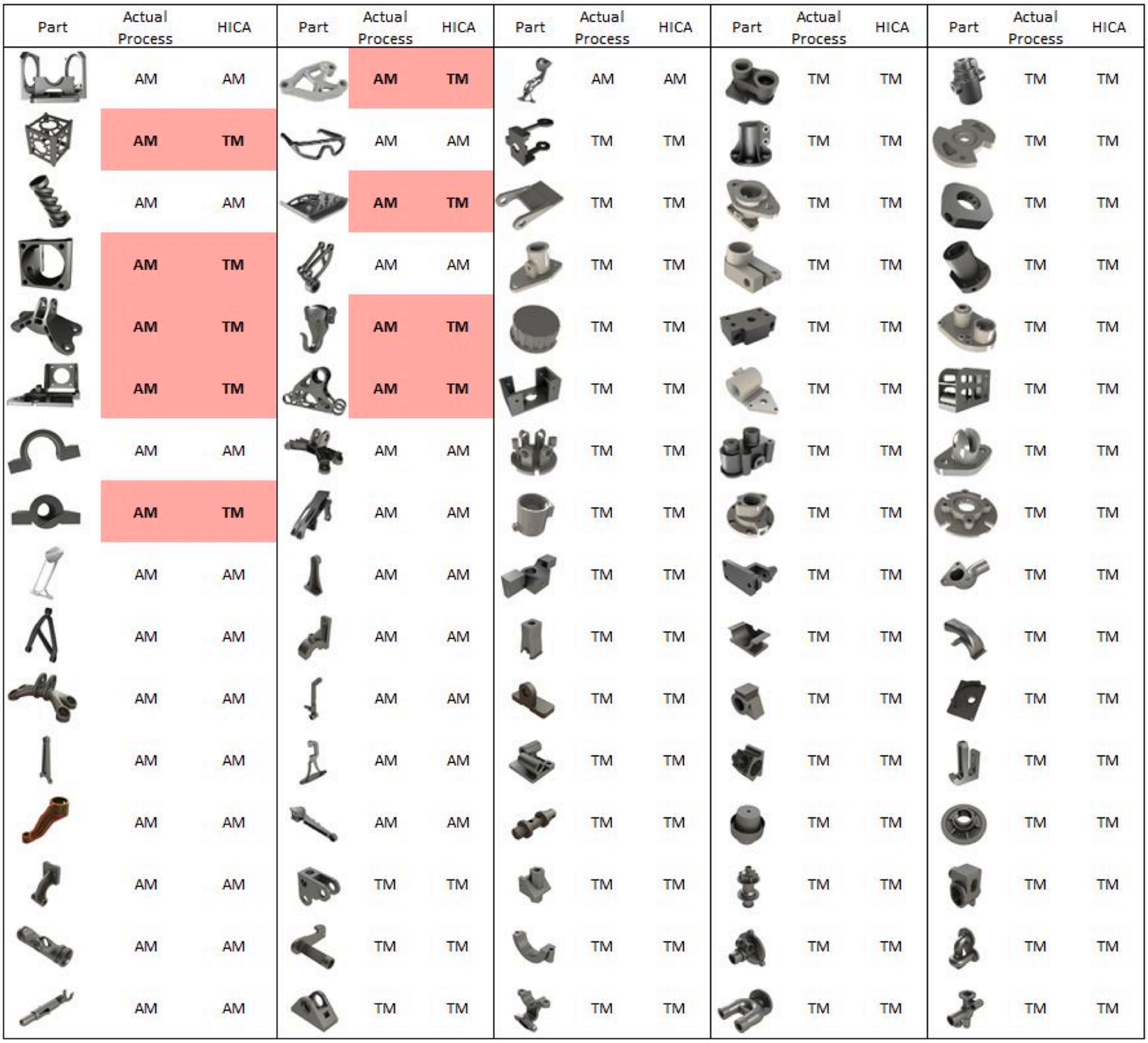
The functions supported by the HICA algorithm without any need to label data, namely the capability of subdividing models or images into clusters, appear as potentially useful for other decision-making activities concerning manufacturing processes. Image clustering or classification has, indeed, a wide range of applications in the manufacturing industry. This makes HICA development and further evolution an opportunity to aid teams in multiple tasks and represents a further motivation for this research work. The following is a list of possible applications based on the concept of HICA and that pertain to the manufacturing domain.

- Engineering design: in the phase of detailed design, engineers can use the HICA to explore the potential of using AM as the production process for developed parts. In a positive case, they can evaluate the opportunity to make shapes even more complex by using, e.g., topology optimization and lattice structures.
- Production monitoring: the algorithm can be integrated into an online production monitoring system that avoids faulty products during production.
- Quality inspection: the HICA algorithm can be used to differentiate faulty products from the production batch after the production.

More concretely, a possible application of HICA in the manufacturing industry could be quality control for parts including welds. Weld images can be differentiated into two clusters based on their external geometries. The HICA algorithm can differentiate the faulty welds and successful welds by clustering weld images into two groups. For example, weld quality was investigated using melt pool images in Xia et al. [22]. These melt pool images can be used as inputs in the HICA algorithm to divide the melt pool images into two categories, which can be used to decide whether the weld was sufficiently accurate or not.

5. Discussion

In the literature, the pros and cons of AM are seen as being at odds with TM processes, which is one of the reasons behind the use of HICA algorithm to differentiate, in a first instance, parts suitable for AM and TM. Nevertheless, it is recognized that other factors affect MPS too. For example, AM includes a number of technologies with different focuses, materials, and properties. In addition, process parameters, such as build orientation and layer thickness can be critical for the success of the manufacturing operation and ultimately crucial for making AM a viable manufacturing technology. It is evident, moreover, that foundry operations such as injection moulding or investment casting are most suitable when a part has to be produced in a large batch and simultaneously presents a relatively complex geometry. In this respect, the subdivision of parts in two clusters made by HICA can be valuable also in those cases where some parameters, e.g., mechanical properties, surface finish or batch volume, make the choice of AM disadvantageous. In these cases,



Part	Actual Process	HICA	Part	Actual Process	HICA	Part	Actual Process	HICA	Part	Actual Process	HICA	Part	Actual Process	HICA
	AM	AM		**AM**	**TM**		AM	AM		TM	TM		TM	TM
	AM	**TM**		AM	AM		TM	TM		TM	TM		TM	TM
	AM	AM		**AM**	**TM**		TM	TM		TM	TM		TM	TM
	AM	**TM**		AM	AM		TM	TM		TM	TM		TM	TM
	AM	**TM**		**AM**	**TM**		TM	TM		TM	TM		TM	TM
	AM	**TM**		**AM**	**TM**		TM	TM		TM	TM		TM	TM
	AM	AM		AM	AM		TM	TM		TM	TM		TM	TM
	AM	**TM**		AM	AM		TM	TM		TM	TM		TM	TM
	AM	AM		AM	AM		TM	TM		TM	TM		TM	TM
	AM	AM		AM	AM		TM	TM		TM	TM		TM	TM
	AM	AM		AM	AM		TM	TM		TM	TM		TM	TM
	AM	AM		AM	AM		TM	TM		TM	TM		TM	TM
	AM	AM		AM	AM		TM	TM		TM	TM		TM	TM
	AM	AM		TM	TM		TM	TM		TM	TM		TM	TM
	AM	AM		TM	TM		TM	TM		TM	TM		TM	TM
	AM	AM		TM	TM		TM	TM		TM	TM		TM	TM

Fig. 4. The comparison between the actual manufacturing processes and the results of HICA algorithm.

the parts belonging to the AM cluster could be favourably thought as candidates for casting. This means that, while HICA has been initially developed as a tool to preselect geometries that exploit the design freedom enabled by AM, its employment can make sense also when parts designated for TM are processed. It is also noteworthy that the subdivision of geometries into two or more clusters foreseen by HICA lends itself to different possible human interpretations: HICA is intended as a ready-to-use support to human decision-making rather than a substitute. In MPS, frequent cases occur where parts can be nearly equally suitable for fabrication with multiple manufacturing technologies. This is because it is hard to clearly establish a demarcation line between conditions that lead to prefer one manufacturing process over another. This fuzzy demarcation line is what makes the development of decision-making approaches relevant for the industry; here, the role of HICA can be seen in the consideration of geometric aspects as a further criterion in the MPS, which is otherwise frequently overlooked.

The usability of HICA in different situations is supported by the outcomes shown in Section 3. To further highlight HICA potential, its classification performance has been compared with the performances of alternative ML algorithms described in the literature, as shown in Fig. 5. In the majority of the articles, SL algorithms, including deep neural networks [7,13], recurrent neural network [12], random forest [6], and support vector machines [14] were used for MPS, which contrasts with the choice of UL-based hierarchical clustering in the current research. The HICA algorithm performed similarly to other available SL-based MPS proposals, but with the clear advantage that HICA does not require prior knowledge, i.e., data labelling, and exploits the only element that is always known in each manufacturing process, i.e., a model or representation showing the geometry of a part.

To evaluate the quality of the clustering achieved by HICA, the Silhouette score is calculated. This score measures the similarity of an object to its cluster; it ranges from −1 to 1, with a higher value indicating better cluster separation. The Silhouette score of the HICA algorithm is calculated as 0.67, which indicates that the clusters generated by the HICA algorithm are well-separated and clearly distinguishable. This score indicates high-quality clustering results, meaning that the data points in each cluster are consistent and well-separated from points in other clusters.

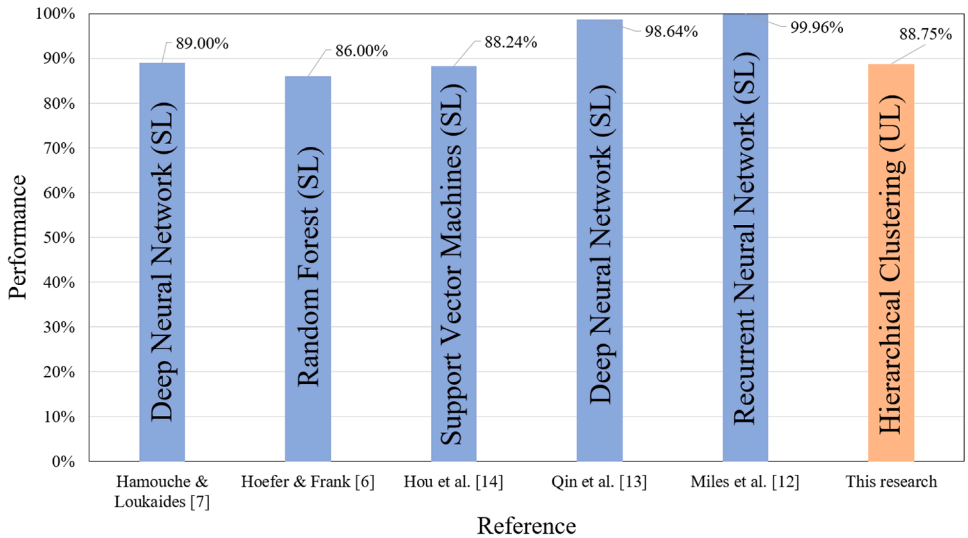


Fig. 5. The performance comparison of the HICA with the literature [6,7,12–14].

HICA is not exempt from limitations, and it is essential to acknowledge them, as they inevitably affect the generalizability and scope of the findings. The following points are highlighted as limitations to the HICA algorithm.

- The proposed software application requires MATLAB and CAD processing software. A basic MATLAB installation with a license to install the "Statistics and Machine Learning Toolbox, Computer Vision Toolbox, and Image Processing Toolbox" is necessary.
- The HICA algorithm needs to be re-run whenever new data is introduced. This can limit the usability of the current HICA platform for other image clustering tasks to be possibly performed in relation to the manufacturing domain; see the bullet list of Section 4.
- Although it is necessary to consider various manufacturing properties to select a manufacturing process, HICA considers part geometry only. This limitation derives from the authors' orientation towards UL and can explain the reduced classification performances of HICA with respect to the most successful SL algorithms shown in Fig. 5.
- Using one isometric view per part can result in omitting some features. Nevertheless, using a single image allows reducing processing and eliminating the need for high computational power.
- Human knowledge is needed to distinguish the two clusters; for example which one includes the parts overall more suitable for AM. As illustrated in the next section, this aspect is a primary trigger for future work.

6. Conclusions and future work

This paper presents a MATLAB-based software tool that differentiates parts appropriate for fabrication with AM and TM technologies by considering their geometry to support decision-making in MPS. This tool uses a UL-based hierarchical clustering algorithm that utilizes unlabelled datasets to differentiate parts. In the provided clustering example, the HICA algorithm performed relatively well, considering its unsupervised nature. The performance of the HICA was measured by comparing the clustering results with the actual manufacturing processes suggested or inferable for 80 CAD models freely available in web repositories. The HICA algorithm correctly classified 88.75 % of the parts. This proves the relevance of part geometries, which is sometimes neglected, and UL in manufacturing operations.

Future work will be primarily earmarked to auto-labelling the resulting clusters, so that the identification of AM and TM clusters will not need human intervention. A further investigation will be conducted to understand the human reasoning behind the association between part geometry and MPS, so that so-far tacit knowledge will be integrated into the algorithm with the aim to improve HICA performances.

CRediT authorship contribution statement

Baris Ördek: Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Data curation. **Yuri Borgianni:** Writing – review & editing, Supervision, Project administration, Data curation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The github link is provided in the metadata section of the main text.

References

[1] Thompson MK, et al. Design for additive manufacturing: trends, opportunities, considerations, and constraints. CIRP Ann 2016;65(2):737–60. https://doi.org/10.1016/J.CIRP.2016.05.004.
[2] Borgianni Y, Pradel P, Berni A, Obi M, Bibb R. An investigation into the current state of education in design for additive manufacturing. J Eng Des 2022;33(7):461–90. https://doi.org/10.1080/09544828.2022.2102893.
[3] Ördek B, Borgianni Y. Differentiating additive and traditional manufacturing processes through unsupervised learning and image processing. In: Borgianni Y, Matt DT, Molinaro M, Orzes G, editors. Towards a smart, resilient and sustainable industry. Cham: Springer Nature Switzerland; 2023. p. 552–63. https://doi.org/10.1007/978-3-031-38274-1_46.
[4] Wang Z, Rosen D. Manufacturing process classification based on heat kernel signature and convolutional neural networks. J Intell Manuf 2022:1–23. https://doi.org/10.1007/S10845-022-02009-9.
[5] Zhang Y, Xu Y, Bernard A. A new decision support method for the selection of RP process: knowledge value measuring. Int J Comput Integr Manuf 2014;27(8):747–58. https://doi.org/10.1080/0951192X.2013.834474.
[6] Hoefer MJ, Frank MC. Automated manufacturing process selection during conceptual design. J Mech Des, Trans ASME 2018;140(3). https://doi.org/10.1115/1.4038686.
[7] Hamouche E, Loukaides EG. Classification and selection of sheet forming processes with machine learning. Int J Comput Integr Manuf 2018;31(9):921–32. https://doi.org/10.1080/0951192X.2018.1429668.
[8] Dohale V, Gunasekaran A, Akarte M, Verma P. An integrated Delphi-MCDM-Bayesian network framework for production system selection. Int J Prod Econ 2021;242. https://doi.org/10.1016/j.ijpe.2021.108296.

[9] Ördek B, Borgianni Y, Coatanea E. Machine learning-supported manufacturing: a review and directions for future research. Prod Manuf Res 2024;12(1):2326526. https://doi.org/10.1080/21693277.2024.2326526.
[10] Li S, Corney J. Multi-view expressive graph neural networks for 3D CAD model classification. Comput Ind 2023;151:103993. https://doi.org/10.1016/J.COMPIND.2023.103993.
[11] Dai W, Erdt M, Sourin A. Self-supervised pairing image clustering for automated quality control. Vis Comput 2022;38(4):1181–94. https://doi.org/10.1007/s00371-021-02137-y.
[12] Miles V, Giani S, Vogt O. Recursive encoder network for the automatic analysis of STEP files. J Intell Manuf 2022;34(1):181–96. https://doi.org/10.1007/S10845-022-01998-X.
[13] Qin FW, Li LY, Gao SM, Yang XL, Chen X. A deep learning approach to the classification of 3D CAD models. J Zhejiang Univ 2014;15(2):91–106. https://doi.org/10.1631/JZUS.C1300185.
[14] Hou S, Lou K, Ramani K. SVM-based semantic clustering and retrieval of a 3D model database. Comput Aided Des Appl 2005;2(1–4):155–64. https://doi.org/10.1080/16864360.2005.10738363.
[15] Liu Y, et al. Graph self-supervised learning: a survey. IEEE Trans Knowl Data Eng 2022. https://doi.org/10.1109/TKDE.2022.3172903.
[16] "MATLAB (2023b)," Natick, Massachusetts: The MathWorks Inc.
[17] Ren H, Li ZN. Object detection using edge histogram of oriented gradient. In: 2014 IEEE international conference on image processing, ICIP 2014; 2014. p. 4057–61. https://doi.org/10.1109/ICIP.2014.7025824.
[18] Gulhane A, Paikrao PL, Chaudhari DS. A review of image data clustering techniques. Int J Soft Comput Eng (IJSCE) 2012;2(1):212–5.
[19] Ördek B, Borgianni Y. Application of unsupervised learning and image processing into classification of designs to be fabricated with additive or traditional manufacturing. Proc Des Soc 2023;3:613–22. https://doi.org/10.1017/pds.2023.62.
[20] Ghosal A, Nandy A, Das AK, Goswami S, Panday M. A short review on different clustering techniques and their applications. In: Jyotsna M, Bhattacharya D, editors. Emerging technology in modelling and graphics. Singapore: Springer; 2020. p. 69–83. https://doi.org/10.1007/978-981-13-7403-6_9.
[21] Saxena A, et al. A review of clustering techniques and developments. Neurocomputing 2017;267:664–81. https://doi.org/10.1016/J.NEUCOM.2017.06.053.
[22] Xia C, Pan Z, Li Y, Chen J, Li H. Vision-based melt pool monitoring for wire-arc additive manufacturing using deep learning method. Int J Adv Manuf Technol 2022;120(1–2):551–62. https://doi.org/10.1007/S00170-022-08811-2.