

Explanations on Smart Stores*

Arturo Acuaviva Huertos, Inmaculada Pérez Garbín, Belén Díaz-Agudo, and
Antonio A. Sánchez-Ruiz

Department of Software Engineering and Artificial Intelligence
Instituto de Tecnologías del Conocimiento
Universidad Complutense de Madrid, Spain
{arturoacuaviva, inmaperezgarbin}@gmail.com
{belend, antsanch}@ucm.es

Abstract. Nowadays, it is common to encounter artificial intelligence in most automated decision-making systems. A great example is the recent introduction by many companies of smart stores that completely automated the checkout process, e.g. Amazon Go or Tao Cafe. However, although there is a clear call for a better understanding on how automated decisions are taken, there is no research available regarding explanations on smart stores to the best of our knowledge. In this paper, we address this concern by introducing a simple and effective way to generate human-understandable explanations for different smart store users. To achieve this, we propose a CBR system which reuses templates of explanations that are retrieved and adapted to match new situations.

Keywords: Explainable AI, Smart Stores, Case-based Reasoning

1 Introduction

In recent years, Artificial Intelligence for decision-making has been on the rise, and many concerns have been raised. In fact, the European Union released, in 2018, the General Data Protection Regulation (GDPR) [1], in which they stated the right of the users to obtain human intervention and contest the decisions made by automated systems. Although the regulation is itself quite controversial [2–4], there is a clear call for more clarity, transparency and disclosure for the algorithms that will be impacting our lives.

Smart stores are a great example of how Artificial Intelligence for automated decisions is shaping the world. A smart store is a retail store that combines smart devices – such as smart shelves or smart carts – to reduce costs, enhance productivity and improve the efficiency in both organizational processes and selling activities [5]. With an increasing number of companies proposing and implementing their own smart store concepts, such as Amazon Go [6] or Tao

* The research leading to these results/this publication has received funding from the iSee project from the CHIST-ERA programme under grant agreement CHIST-ERA-19-XAI-008.

Cafe [7], a new field is clearly emerging. Despite this fact, even so there are quite a few papers introducing smart retailing concepts or ideas [8–13]; most of them introduce unmanned, unstaffed or cashier-less stores without diving deep into the need for explanation. Besides, this necessity is even greater for unmanned stores, which completely automate the in-store shopping experience.

This need for explanation can be met at two different levels, derived from the fact that most of these establishments work by leveraging sensor fusion and computer vision: explain the system as a whole, or clarify each concrete algorithm or machine learning model implemented. Note that most of the techniques applied for product and customer recognition, and tracking, involve dealing with deep neural network models, and, consequently, any prediction can involve millions of mathematical operations. For this reason, despite the complexity of deep learning models is a reason for their increased performance and accuracy, their nested non-linear structure makes them highly non-transparent (i.e. it is not clear what information in the input data makes them actually arrive at their decisions). Therefore, these models are typically regarded as black boxes [14]. Fortunately, black boxes models are being extensively studied and many methods have already been proposed to try to explain them [15]. In consideration of the foregoing, we want to focus not on explaining the different deep learning models that made up an unmanned smart store, but on clarifying the inner working of the retail establishment itself. Indeed, this is a concern that has not been addressed yet in the scientific literature to the best of our knowledge.

Explanations on smart stores, as a whole, can be of great interest to customers, store managers, and any staff member, since they fulfill the need of understanding how automated decisions are taken. In fact, customers can benefit from explainable smart stores by being able to request a clarification on why a certain product was charged to their account. Most customers will request this clarification whenever they are certain that a product was wrongly added to their virtual shopping cart. Reviewing the explanation provided, they can then decide whether to agree on the decision made by the automated store or challenge it by justifying why the predictions of the system failed (e.g they can argue that although they took the product, they later put it back). This process does not only provide a way for customers to solve their issues, but also highlights the failures of the system so they can be addressed. On the other hand, store managers can obtain insights on past events for the store and receive more technical details on why the system behaved the way it did. This information is critical to ensure that any error is detected as early as possible (e.g. the system could notify that, because of a crowded environment, occlusion prevent the smart store from operating reliably). All in all, explanations do not only shed light on the decisions, but also build trust in the automated smart systems.

In the following sections, we detail the general architecture for unmanned smart stores to be able to operate, as well as introduce the different types of users and the explanations each one demands. In addition to this, we introduce how Case-Based Reasoning (CBR) can be used to provide explanations in a smart store, and we show how to deal with some concrete case scenarios. Finally, we

discuss the questions arising from the results of this work and expose our main conclusions.

2 Unmanned smart stores architecture

Although there is no public and accessible information, nor research documents, showing the inner working of unmanned smart stores, companies have been claiming to use different technologies in their recent proof of concepts [7, 11, 16, 17]. For this reason, we outline the most common techniques and technologies claimed to be implemented in these retail establishments. This information is particularly useful to describe the general context in which we assume to be operating, and, therefore, to know what data will be available to generate the explanations.

The first simple device that is commonly found in smart stores is a smart shelf. Smart shelves can be conceived as the natural technological evolution of conventional racks. A simple way to create this device is to use load/weight sensors to measure the quantity of the items available. However, there have been some alternatives proposed that extend this model. Some of them are based on computer vision approaches to quantify if the product is in the right position (i.e. on the right shelf) or placed with right visibility to the shoppers [18]. Mainly, we will assume that there are smart shelves available which tell us which items are available and where are they placed at any time in the store.

On the other hand, a frequent technique used in smart stores is deep learning for computer vision, since it is useful to deal with tasks such as assessing who takes an item, tracking customers inside the store, or recognizing in-store products. In particular, we want to highlight the following problems that are usually addressed:

- **Object Detection.** The process of detecting instances of objects of a certain class within an image is named object detection. Detecting instances can be of great interest in areas such as video surveillance, customer tracking inside a store, or product recognition. This information combined with human tracking is critical, since it can be used to identify customers who are taking or dropping a certain object. Additionally, it can also be useful to recognize items placed in the shelves.
- **Multitarget Tracking.** Human tracking is the task of estimating the location of a moving person through a set of consecutive frames of images. Moreover, the problem of estimating a moving object is not limited to human tracking, but it is generalized to multiple object tracking or multiple target tracking [19]. Tracking algorithms are suitable for smart stores since they offer a way of monitoring customers moving around in the establishment. This information can later be used to estimate who took or dropped an item to a shelf, or notify when a user enters or leaves the store. Apart from helping to figure out who performed a certain action that changed the state of the store, people tracking provides meaningful data for marketing analysis that can drastically improve targeted advertising.

- **Pose Estimation.** Given an image, human pose estimation is the process of determining the positions of human body parts such as the head, hands, wrists, knees or elbows [20]. This technique plays a major role for almost every smart store model, since this estimation helps the computer to assess the best candidate who performed an interaction with the store, such as grabbing or dropping a product.

3 Explanations for different types of users

There are mainly two different types of users in smart stores: customers and staff. We will now briefly describe each profile and introduce our explanation proposals for each one.

3.1 Customers

Trusting the decisions made by the automated store is a major concern for customers. Whereas brick-and-mortar establishments offer an explicit checkout process that ensures that the only products considered are the ones taken by the customers, unmanned smart stores completely hide the payment process. Instead, the decision on what products will be charged is automated, relying on the data gathered from the sensor fusion network to make it. In this way, they could be mostly interested in knowing why a certain product was added to their account, so that they can review the decision.

According to the context provided above, we consider that most customers could greatly benefit from visual explanations. A video of the exact moment when the customer took an item would justify the decision taken. Additionally, if the customer reviews the video and finds out that another person took the product, or they later put back the product but the video did not show it, they can ask for human intervention to contest the decision.

In order to ensure that the smart store respect the right to privacy of the users, the system could blur all other customers in the video scene using information from the position tracked for each customer. Cropping and zooming the video to focus on the area of the shelf that contained the object could also help to provide a more targeted explanation, as well as trimming the video to include just the exact moment when the item was taken.

3.2 Staff members

Store staff is mostly concerned about the proper functionality of the system. In this regard, they would be interested in detecting any potential issue that might lead to a poor shopping experience for customers. On the one hand, in brick-and-mortar retailers, the staff can actively supervise how customers are interacting with the store, while they keep records of the purchases made. On the other hand, in an unmanned smart store, the store staff can be reduced to just store managers or store supervisors who want to ensure that the store

is properly and autonomously operating. Hence, whenever a complex situation arises, they want to be able understand why the system behaved the way it did, so they can evaluate the situation and decide whether any fix or modification is required.

In consideration of the previous statements, the unmanned smart store must be able to provide explanations which clarify why the system made a certain decision, providing technical details and insights. We propose an approach based on the 4R of the Case-based Reasoning (CBR) cycle [21] to be able to generate explanations to different in-store events. The main idea is that the smart store will provide explanations for a certain in-store occurrence by retrieving the most similar case. Then, it will reuse the information and knowledge acquired in the past to try to explain the new case. After that, it will revise the solution proposed and retain new parts that could be of interest when classifying new future events. In the following sections, we will go through this CBR explanation approach in depth.

4 CBR for explanations

As we have previously exposed, CBR can be advantageous when trying to generate explanations for staff members in the store. In this section we introduce how we represent each case and their solutions provided to the customers in the form of filled up templates. Then, we move to detail each step of the 4R cycle.

4.1 Case representation

Since most of the information available from the sensor fusion network can be encoded numerically the most appropriate approach is to use a feature-vector representation. This technique represents a case as a vector of attribute-value pairs. In this setup, the cases are sometimes known as propositional cases and they support k -nearest neighbor matching and instance-based learning [22]. Moreover, this representation also supports distance-based similarity metrics and common techniques related to Information Retrieval.

In a particular unmanned smart store scenario, each case represents an action that changed the state of the store (i.e. a customer took or drop an item to the shelves). As we want to explain a certain past prediction for the event, we assume that each customer received a likelihood score of being the one performing the action. Furthermore, we also presume that the human poses estimated for each customer received a certain prediction score. More detail on what data is gathered and available in the smart retail setup can be found in the literature [10–12].

Considering the statements above, we can use the following attributes to build the feature-vector case:

- The number of customers close to the shelf, n .
- The likelihood of customers who scored the highest value, s_1 .

- The likelihood of customers who scored the second highest value, s_2 .
- The pose estimated score prediction for the customer who scored the highest likelihood¹, p_1 .
- The pose estimated score prediction for the customer who scored the second highest likelihood¹, p_2 .
- The number of customers who scored the highest likelihood of performing the action, m_1 .
- The number of customers who scored the second highest likelihood of performing the action, m_2 .
- The distance from the highest scoring customers to the shelf, d_1 .
- The distance from the second highest scoring customers to the shelf, d_2 .
- The shelf impacted by the action, s .
- The object taken or dropped, o .
- The description of the action, whether it was an object taken or dropped, a .

Using these attributes, we can build a case i general vector as it follows:

$$C_i = (n, s_1, s_2, p_1, p_2, m_1, m_2, d_1, d_2, s, o, a) \quad (1)$$

Since we will be working with vectors of different lengths (i.e. C_i which might lack any of the attributes introduced before), we denote A_i to the set of all attributes of the i -th case. Moreover, we name each attribute $\alpha \in A_i$ of a C_i vector case as C_i^α . Finally, we will not use the attributes s , o , and a to make predictions; therefore, we denote this subset of attributes as A' . To simplify the notation in this work, we consider that any set of attributes A_i is used after subtracting the elements of A' , so that none of the elements of A' are included in A_i .

4.2 Case solution

The explanations of each case are known as the case solutions, and they are based on templates dynamically adapted to each case. The adaptation process approach is similar to the one taken by Diaz-Agudo et al. [23]. In this sense, each retrieved case has an associated template that includes text information describing the scene as well as the attributes of the case.

In general, the template is divided into three sections: the header (a compilation of parameters that define the case), the body (a text explanation) and the footer (the similarity score). A generic template is detailed at Figure 1.

Note that, although the generic template portrayed contains all of the parameters, a concrete template to explain a certain case might be limited to a subset of attributes.

¹ In case there is more than one customer with the same probability, we consider the mean value of their pose estimated score predictions.

ATTRIBUTES	
<i>Description</i>	<i>Value</i>
Number of customers close to shelf	n
Likelihood of highest value customers	s_1
Likelihood of second highest value customers	s_2
Pose estimated score prediction (highest value customers)	p_1
Pose estimated score prediction (second highest value customers)	p_2
Number of customers scoring the highest likelihood	m_1
Number of customers scoring the second highest likelihood	m_2
Distance from highest scoring customer to shelf	d_1
Distance from second highest scoring customer to shelf	d_2
Shelf impacted by the action	s
Object (taken or dropped)	o
Action (take or drop)	a
EXPLANATION	
<i>Text information containing the clarification</i>	
<i>Similarity Score</i>	α

Fig. 1: Generic template associated to a case

4.3 Retrieval

We propose a retrieval constraint-based methodology for filtering previous cases and avoid over-generalization [24], and an estimation of similarity based on the nearest-neighbour (NN) matching function [25]. Our approach extends the one introduced by Montazemi and Gupta [26]. Although we use the same constraints acquisition process to acquire the validity constraints from the domain expert, we modify the matching function not to compute similarities along any descriptors but to calculate the similarity of a subset of numerical values. The validity constraints will later be useful to filter out cases that are not applicable for the incoming query (e.g. when the human pose prediction scores are really low, cases that describe the event using information such as the action likelihood for a certain customer might not apply).

Regarding the similarity, we introduce now the likelihood function that we propose to be used for the retrieval of similar cases:

$$\mathcal{L}(C_i, C_k) = \frac{\sum_{j \in A_i \cap A_j} w_j^k \cdot \text{sim}(C_i^j, C_k^j)}{\sum_{j \in A_i \cap A_j} w_j^k} \quad (2)$$

where C_i is the new case and C_k a past case we are comparing against, and A_i and A_j are their respective set of attributes. The values w_j^k are called *weights* of the attribute j for the k past case, and they are a measure of the importance of the attribute. Additionally, the function $sim(C_i^j, C_k^j)$ compares the attribute j instance for the new case C_i and the past case C_k . In order to compute the function $sim(C_i^j, C_k^j)$, we propose a strategy which considers the maximum value M_j that any attribute j can take, and then we can express the function as it follows:

$$sim(C_i^j, C_k^j) = 1 - \frac{|C_i^j - C_k^j|}{M_j} \quad (3)$$

As for the weights w_j^k , we considered them as contextual-importance values which help discriminating among previous cases. To quantify these values, we compute the variation of the standard deviation σ_j of the attribute j for the d previous candidates analyzed:

$$\sigma_j = \sqrt{\frac{1}{d} \sum_{k=1}^d (C_k^j - \bar{\mu}_j)^2} \quad \text{where} \quad \bar{\mu}_j = \frac{1}{d} \sum_{k=1}^d C_k^j \quad (4)$$

Hence, the importance of the j attribute to discriminate cases, with respect to a new case C_i , can be seen as the relative importance of the attribute:

$$w_j^k = \frac{\sigma_j}{\sum_{p \in A_i \cap A_k} \sigma_p} \quad (5)$$

After assessing the new case C_i against past cases, we must decide which case fits better the new instance. In order to achieve this, we take the best similar case according to the output reported by the likelihood function \mathcal{L} . However, there are two impasses that may occur: the situation is unknown because there is no memory available to match the new case, or there are several cases that apply with the same degree of confidence. In both situations, we assume an alarm is triggered so that an expert intervenes to select the best fit among all the options, or to provide a new different explanation.

4.4 Reuse

In the process of reusing the case we have obtained, we retrieve the template linked to the matched case (Section 4.2) and simply replace its values with the values in the query. Since the retrieved case and the query could have different sets of attributes², we distinguish between two situations: the retrieved case and the query have the same set of attributes, or the query has more attributes than the retrieved case. To solve both, the attributes of the retrieved case are filled with the corresponding query values.

² Whereas the query contains every attribute, the retrieved case have a selection of them (i.e. the cardinal of the attributes of the retrieved case is less or equal than the cardinal of the attributes of the query).

ATTRIBUTES	
<i>Description</i>	<i>Value</i>
Number of customers close to shelf	n
Likelihood of highest value customers	s_1
Number of customers scoring the highest likelihood	m_1
Shelf impacted by the action	s
Object (taken or dropped)	o
Action (take or drop)	a
EXPLANATION	
Action a performed on shelf s , containing o . When the event occurred, there were n customers around the shelf. There were m_1 customers scoring the same highest likelihood (s_1). For this reason, the system was prevented from properly assessing who performed the action by occlusion in a crowded environment.	
<i>Similarity Score</i>	α

Fig. 2: Template for occlusion issues

4.5 Revision

Having mapped a past case to the target query, the contribution of an expert in the domain is required to revise it, and no automated solution to this stage of the CBR cycle is contemplated. In order to facilitate this labor, the explanations can be accompanied by a video presenting the scene that is being studied. This video is tagged with information about the pose estimation and tracking of the customers, as well as product recognition labelled. Then, the domain expert proceeds to analyze and evaluate the solution proposed by the system. In this stage, we also contemplate the possibility of receiving feedback to modify the solution proposed, and further adapt an existing solution or derived a new subcase.

4.6 Retention

After the revision, if the case was a new instance of a previously unknown situation or the template of the case was modified, the new case within the filled template is added to the knowledge base as a past case. Our system learns from observations, so we assume that a representative set of cases is provided beforehand. Moreover, an expert can provide a new case learnt from direct observation at any time. In this regard, the revision stage is particularly important to notice any new case instance that must be addressed, whenever it has not been properly identified.

5 Example scenarios

We now introduce some different case scenarios that may occur, and its classification using the templates shown before.

ATTRIBUTES	
<i>Description</i>	<i>Value</i>
Number of customers close to shelf	1
Shelf impacted by the action	s
Object (taken or dropped)	o
Action (take or drop)	a
EXPLANATION	
Action a performed on shelf s , containing o . When the event occurred, there was only one customer around the shelf. For this reason, the system directly identified that customer as the source of the action.	
Similarity Score	α

(a) Template for single customer in store

ATTRIBUTES	
<i>Description</i>	<i>Value</i>
Number of customers close to shelf	n
Likelihood of highest value customers	s_1
Likelihood of the second highest value customers	s_2
Number of customers scoring the highest likelihood	1
Shelf impacted by the action	s
Object (taken or dropped)	o
Action (take or drop)	a
EXPLANATION	
Action a performed on shelf s , containing o . When the event occurred, there were n customer around the shelf. There was just one customer scored with the highest likelihood value (s_1). As it is much greater than the second highest score s_2 the system identified that customer as the source of the action.	
Similarity Score	α

(b) Template for multiple customers in store

Fig. 3: Templates for actions of customers

5.1 Occlusion in crowded shelf area

A common situation that occurs in any retail establishment is the presence of large crowds in a relatively small space. That situation could lead the service to lose reliability in the data provided by the sensor fusion network. Moreover, when many customers are grouped in a certain small area, it is not possible to avoid having issues such as theft offenses. This concern does not only impact smart stores but brick-and-mortar retail establishments too.

If a staff member is interested in understanding why the system took a certain decision in this situation, without knowing the context, then the CBR system could provide a clarification on this. Figure 2 introduces the solution template for a generic case that can be related to occlusion issues in crowded environments, which led to poor smart store estimations.

5.2 Customer performing an action

We introduced here two simple events that are triggered by customers when taking or dropping an object. Firstly, we consider the case when a single customer is in the store. Hence, the system could effortlessly detect the person performing the action. Figure 3a shows the solution template for this elementary case. Secondly, we can consider the case when multiple customers are in the store, but only one customer scored a high value in comparison with the others. In this case, it is quite patent who performed the action (see Figure 3b for details).

6 Conclusions

Our aim in this paper has been to introduce a feasible strategy to take the edge off the lack of explainability models for unmanned smart stores. Consequently, we tried to expose how visual explanations using videos, along with textual descriptions, can be helpful when providing explanations for smart retail establishment users. Particularly, we proposed a CBR-based system to generate explanations for staff managers. These human-understandable explanations seek to streamline the inner working of complex systems, which leverages data from different sensors and combines several deep learning techniques. In this sense, our proposal abstracts features that can be useful to define the overall behavior of the store, without having to worry about each particular component that is used to build it. Although an explainability system has been fully defined in this work, further research is required in the field of explainability for smart stores. For this reason, we propose several ways to proceed with future work in this area. In particular, the model proposed here can be implemented to match more complex scenarios, in which more data from the sensor fusion network is available. In addition to this, cases could be modified, so that they can be classified according to the risk of misbehaviour of the model when assessing the actions of the customers.

References

1. *European General Data Protection Regulation* European Commission. https://ec.europa.eu/info/law/law-topic/data-protection_en (2021).
2. Edwards, L. & Veale, M. Slave to the Algorithm? Why a Right to Explanation is Probably Not the Remedy You are Looking for. *SSRN Electronic Journal* **16**. DOI: 10.2139/ssrn.2972855 (2017).
3. Malgieri, G. Automated decision-making in the EU Member States: The right to explanation and other “suitable safeguards” in the national legislations. *Computer Law & Security Review* **35**, 105–327. ISSN: 0267-3649. DOI: <https://doi.org/10.1016/j.clsr.2019.05.002> (2019).
4. Wachter, S., Mittelstadt, B. & Floridi, L. Why a Right to Explanation of Automated Decision-Making Does Not Exist in the General Data Protection Regulation. *International Data Privacy Law* **7**, 76–99. ISSN: 2044-3994. DOI: 10.1093/idpl/ix005 (2017).
5. Pantano, E. & Timmermans, H. What is Smart for Retailing? *Procedia Environmental Sciences* **22**, 101–107. ISSN: 1878-0296. DOI: <https://doi.org/10.1016/j.proenv.2014.11.010> (2014).
6. Amazon. *Amazon Go* <https://www.amazon.com/-/es/b?ie=UTF8&node=16008589011> (2021).
7. Tan, Y. *China’s version of Amazon’s cashier-less store is here* 2017. <https://mashable.com/2017/07/12/taobao-china-store/?europa=true\#570XUGTQVaag> (2021).
8. Guo, B. *et al.* DeepStore: Understanding Customer Behaviors in Unmanned Stores. *IT Professional* **22**, 55–63. DOI: 10.1109/MITP.2019.2928272 (2020).
9. Pantano, E. & Dennis, C. *Smart Retailing: Technologies and Strategies* ISBN: 978-3-030-12607-0. DOI: 10.1007/978-3-030-12608-7 (2019).

10. Xu, J. *et al.* Design of Smart Unstaffed Retail Shop Based on IoT and Artificial Intelligence. *IEEE Access* **8**, 147728–147737. DOI: 10.1109/ACCESS.2020.3014047 (2020).
11. Liu, X., Jiang, Y., Kim, K.-H. & Govindan, R. *Grab: Fast and Accurate Sensor Processing for Cashier-Free Shopping* 2020. arXiv: 2001.01033 [cs.CV].
12. Ruiz, C., Falcao, J., Pan, S., Noh, H. Y. & Zhang, P. *Autonomous Inventory Monitoring through Multi-Modal Sensing (AIM3S) for Cashier-Less Stores in Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation* (Association for Computing Machinery, New York, NY, USA, 2019), 395–396. ISBN: 9781450370059. DOI: 10.1145/3360322.3361018. (2021).
13. Wang, Y.-C. & Yang, C.-C. 3S-cart: A Lightweight, Interactive Sensor-Based Cart for Smart Shopping in Supermarkets. *IEEE Sensors Journal* **16**, 6774–6781. DOI: 10.1109/JSEN.2016.2586101 (2016).
14. Samek, W., Wiegand, T. & Müller, K.-R. Explainable Artificial Intelligence: Understanding, Visualizing and Interpreting Deep Learning Models. *ITU Journal: ICT Discoveries - Special Issue 1 - The Impact of Artificial Intelligence (AI) on Communication Networks and Services* **1**, 1–10 (2017).
15. Guidotti, R. *et al.* A Survey of Methods for Explaining Black Box Models. *ACM Comput. Surv.* **51**. ISSN: 0360-0300. DOI: 10.1145/3236009 (Aug. 2018).
16. Amazon. *Just Walk-Out Technology* 2020. {<https://justwalkout.com/>} (2021).
17. NTT Data. *Announcing the introduction of Facial Authentication and Dynamic Pricing at Catch&Go™ digital grocery stores* <https://www.nttdata.com/global/en/media/press-release/2020/february/announcing-the-introduction-of-facial-authentication-and-dynamic-pricing> (2021).
18. Satapathy, R., Prahlad, S. & Kaulgud, V. *Smart Shelfie – Internet of Shelves: For Higher On-Shelf Availability in 2015 IEEE Region 10 Symposium* (2015), 70–73. DOI: 10.1109/TENSYMP.2015.9.
19. Luo, W. *et al.* Multiple object tracking: A literature review. *Artificial Intelligence* **293**, 103448. ISSN: 0004-3702. DOI: <https://doi.org/10.1016/j.artint.2020.103448>. (2021) (2021).
20. Ouyang, W., Chu, X. & Wang, X. *Multi-source Deep Learning for Human Pose Estimation in 2014 IEEE Conference on Computer Vision and Pattern Recognition* (2014), 2337–2344. DOI: 10.1109/CVPR.2014.299.
21. Aamodt, A. & Plaza, E. Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. *AI Commun.* **7**, 39–59. ISSN: 0921-7126 (Mar. 1994).
22. Bergmann, R., Kolodner, J. & Plaza, E. Representation in case-based reasoning. *Knowledge Eng. Review* **20**, 209–213. DOI: 10.1017/S0269888906000555 (Sept. 2005).
23. Belén Díaz-Agudo, J. A. R.-G. & Jimenez-Díaz, G. Data explanation with CBR. *ICCBR 2018*, 64–74 (2018).
24. Eskridge, T. Principles of continuous analogical reasoning. *Journal of Experimental & Theoretical Artificial Intelligence* **1**, 179–194. DOI: 10.1080/09528138908953700 (1989).
25. Duda, R. *Pattern classification and scene analysis* ISBN: 0471223611 (Wiley, New York, 1973).
26. Montazemi, A. & Gupta, K. A framework for retrieval in case-based reasoning systems. *Annals of Operations Research* **72**, 51–73. DOI: 10.1023/A:1018960607821 (1997).