A Review of Action Anticipation in Human-Robot Collaboration

Pedro Amaral
DETI
Universidade de Aveiro
Aveiro, Portugal
pedro.amaral@ua.pt

***Abstract*—With the increase in the diversity of products on sale due to the evolution of technology and standard of life, there is a growing demand for flexible manufacturing that can meet the necessary production, especially in small companies. Although the usual solution for these needs is to use human operators, which provide the necessary flexibility and precision, this solution comes at a greater cost. In contrast, industrial robots offer a relatively smaller price and show more value in repetitive and heavy tasks. That is where Human-Robot Collaboration (HRC) comes into action since it complements the flexibility of a human worker with the strength and lower cost of the robotic worker in the same workspace. Inside HRC, Action Anticipation is a technique used in collaborative assembly to predict the actions of the human workers so that the robot can better plan its movements, increasing manufacturing efficiency and safety. This article reviews the research in this field, including the commonly used data sources and algorithms with a particular focus on machine learning methodologies.**

***Index Terms*—Human-Robot Collaboration, Human-Robot Interaction, Action Anticipation, Cobot, Artificial Intelligence, Machine Learning**

I. Introduction

The Third Industrial Revolution was characterized by a focus on automating repetitive and heavy tasks on the assembly lines. Still, this created a problem: whenever the manufacturers needed the robots to work in a different assembly process, they needed to be reprogrammed by an expert.

The Fourth Industrial Revolution, also known as Industry 4.0, refers to the current trend of the manufacturing sector to become more intelligent and achieve greater automation. This trend takes advantage of the recent developments in artificial intelligence, the Internet of Things, and autonomous robots to pave the way for more efficient and flexible production processes. With Industry 4.0, robots are expected to be more adaptable and perform more actions without constant explicit programming.

Human-Robot Collaboration (HRC) aims to achieve greater efficiency by using robots in the same workspace as humans taking advantage of the strong points of both of them. These are called Collaborative Robots or Cobots and have significant benefits when working with people. For once, they can safely work with people since they have sensitive sensors that can detect the human interrupting them, causing them to stop their actions. They are also smaller, compact, and easy to program, among other advantages.[1]

One of the sub-fields of Human-Robot Collaboration is Human Action Anticipation which is the focus of this review.

A. Definition of Anticipation

Anticipating actions is an idea that comes from biology, given that humans and many other animals anticipate each other constantly. In biology research, we can find a definition such as the one stated in [2]: ”An anticipatory system is a system containing a predictive model of itself and/or its environment, which allows it to change state at an instant in accord with the model’s predictions pertaining to a later instant.”.

In Robotics, we can find a similar definition such as in [3]: ”Action anticipation consists of classifying an action even before it occurs, by using the partial information provided up to a certain moment in time.”.

From these definitions, human action anticipation in HRC can be considered as the robot predicting the future actions of the human worker before he does them and then reacting accordingly.

To better visualize the real-world application, suppose, for example, that a human worker needs a specific material, such as a wooden plank. In this case, the robot can anticipate it and either provide it as it did in Fig. 1 or move out of the way to avoid a collision. Furthermore, anticipating human actions helps improve the overall speed and manufacturing efficiency of the collaborative assembly while also helping to reduce the risk of accidents or injuries, increasing safety.

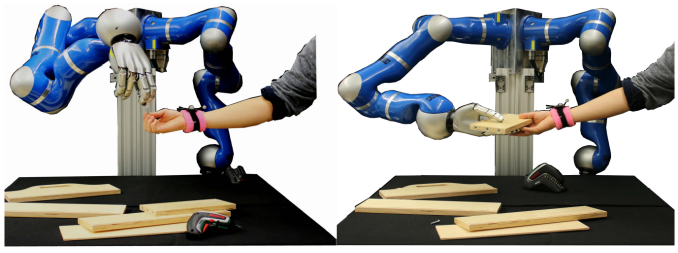

Fig. 1. Situation where a robot anticipates that the worker will need a wooden plank and hands it over to him [4]

B. Article Structure

An action anticipation problem can be divided into three sub-problems. First, it must be decided which data needs to

be captured by the sensors, then which algorithms should be used to analyze the captured data, and finally, how to increase the safety of the user. This division is also used to structure this article.

II. Data Sources and Sensors

The first step to anticipating the following action is to know what kind of data we should collect with the sensors. In this section, a broad overview of the communication methods standard in HRC will be done followed by a more specific reference to those that are applied to action anticipation in the literature. Even if some of the methods listed cannot serve as a data source to perform anticipation, they are still relevant in HRC, and they can be helpful when developing a real implementation.

A. Interaction in HRC

In Fig. 2, we can see a diagram containing multiple data sources that can be used to implement communication between the robot and the human, along with its advantages and disadvantages.

1) Gestures: Gestures are one of the main ways humans communicate, whether through simple movements or formal sign language. In work about Human-Robot Collaboration, gestures can also commonly be found since it has the advantage of resisting ambient noise.

Usually, gestures are captured with vision-based methods with either an RGB or RGB-D camera, so there is no need for unnatural movements. With vision, it is possible to include markers, but these may lead to occlusions and hinder the worker's movements. Consequently, there is also work in the literature that uses markerless vision to allow more unrestricted movements.

Another way to capture the movements of the human worker would be to use wearable inertial sensors, which contain accelerometers and gyroscopes, but, once again, wearables can hinder the worker's movements.

2) Natural Language: Natural Language is the main and the most intuitive way for humans to communicate with each other. The advances in natural language processing make this a possible communication solution with robots. However, despite being intuitive, simple, effective, and even robust against lighting variations, when it comes to an industrial setting that contains significant sound noise, it becomes less valuable than the alternatives.

3) Gaze: Next, the gaze can also be used to determine where the user's attention resides, giving a considerable amount of information that can trigger some action.

There are two options to obtain the user's gaze. Wearable sensors can provide better results but are expensive and intrusive. On the other hand, algorithms that detect head pose and assume the gaze from it can also be used, which is a cheaper and non-intrusive solution.

4) Emotions through Facial Expressions: Although this is a relatively new idea, some applications analyze the user's emotions from his facial expressions to have even more information in the algorithms.

5) Semantics: Finally, semantic information about the objects can also help the global workflow. For example, suppose the robot is trained to recognize certain features in objects related to how it can pick them up. In this case, the robot can pick up a new object it has never seen before if it has a similar structure.

Human actions can also be represented semantically by obtaining the poses of the human as a specific set of limbs, even if only partially. During action recognition, this can be used to know which objects the worker can interact with.

Having information about the pose of the human body also helps in the path-planning phase of the robot since it can use this information to avoid the worker and prevent collisions.

B. Interaction in Action Anticipation

Previously, several forms of communication between humans and robots were described. Still, these work in a more active way, and not all of them can be applied to action anticipation, where the user should not need to do anything for the robot to act. Essentially, there is a need to capture the human's body language.

As humans usually anticipate each other by poses and gestures, these factors became some of the most common data to perform action anticipation. Regarding sensors, most of the literature suggests using an RGB camera. Still, some works, such as the one described in [6], indicate the use of an RGB-D camera to capture both the color and the depth images. If wearable sensors are an option, inertial sensors also become an alternative.

In [4], the authors also used markers to obtain the gestures of the human.

In [7], [8], [9], and [10], the authors went a step further and used the images but also processed the optical flow between them and used it in their algorithms.

In [3], the authors used OpenPose[1][11][12][13][14] which is a framework with pre-trained models that receive an image, process it, and return 3D points representing the skeleton joints of the person in the image, as we can see in Fig. 3.

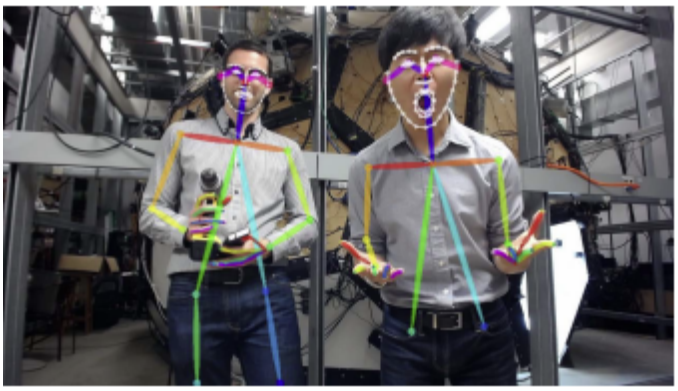

Fig. 3. OpenPose example[11]

[1]OpenPose documentation:
https://cmu-perceptual-computing-lab.github.io/openpose

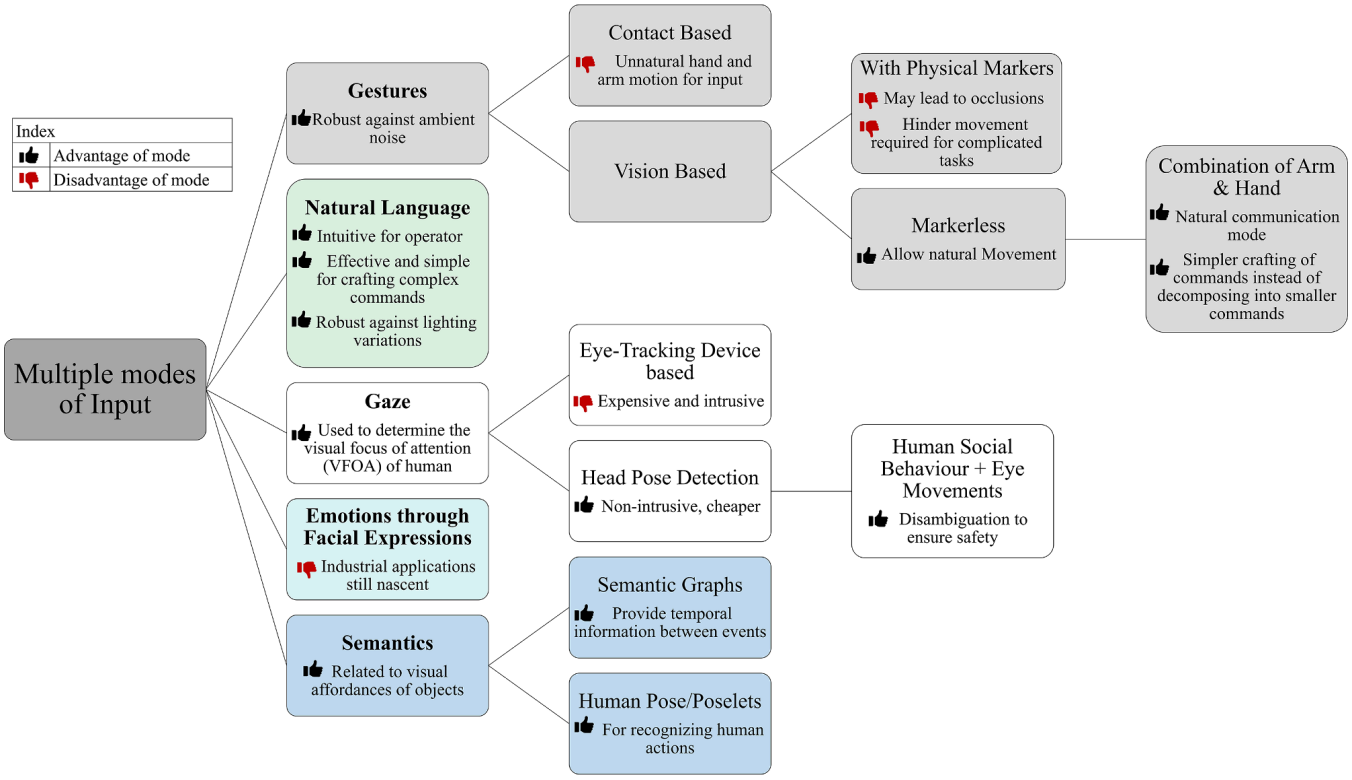

Fig. 2. Advantages and Disadvantages of some Data Sources in Human-Robot Collaboration [5]

Humans also tend to anticipate each other by considering the other's gaze, which usually indicates his center of attention. As this is also an involuntary aspect, there is some work where gaze provides additional information, such as in [15] where the dataset contained the gaze of the user captured with wearable sensors or in [3] where the gaze was assumed from the results of an algorithm to detect the head pose.

In addition to the data related to the human, the objects present in the environment can also give valuable information about the human's following action, as is the case in [10].

III. Algorithms

After knowing which is usually captured and provided to an algorithm, this section explores possible algorithmic solutions present in previous work.

Machine Learning algorithms have been increasingly more common in the last years due to, for example, their ability to deal with multidimensional data. These algorithms can automatically learn from data and make predictions or decisions, which makes them a prime candidate to use in the context of human action anticipation in collaborative environments. The most common strategies in this field are Supervised Learning, Unsupervised Learning, and Reinforcement Learning. As we can see in Fig. 4 obtained from a review article about HRC in general, supervised learning and reinforcement learning are dominant in this area, with composite solutions surpassing unsupervised learning in the most recent year showed.

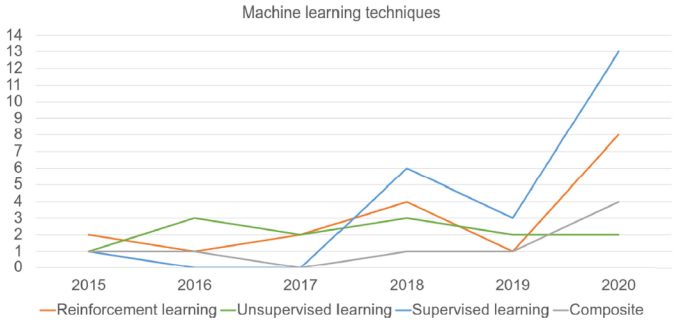

Fig. 4. Number of articles relevant to the HRC review from each machine learning technique throughout the years[16]

Unsupervised learning is more valuable for finding structure in the data, creating clusters based on common characteristics, or identifying anomalies and outliers from unlabeled datasets. Consequently, it is very rare in the action anticipation literature. An example was found in [17] but it was a composite solution. Given that these use cases are very different from those of Action Anticipation, this article will focus on the other types of learning.

A. Supervised Learning

In Supervised Learning, the models are trained using a dataset of labeled data. The models from this group are further divided into classification, where the new instance is assigned a particular class, and Regression, where it is

given a certain real number. These models must generalize the knowledge from the examples to deal with a new instance correctly that they have never seen before. Among these models, convolutional and recurrent neural networks are at the forefront of the algorithms to explore.

A Recurrent Neural Network (RNN) is a type of neural network where the output of each time step is fed back into the input at the next time step, allowing the network to remember and incorporate information from previous time steps into its processing of current and future data. This characteristic makes RNNs particularly well-suited to processing sequential data, such as text, speech, or time series data which require context or temporal dependencies. In particular, LSTM is an RNN with a more complex architecture that gives it an improved ability to backpropagate the error, making it better to train a model that classifies sequences with several time steps.

A Convolutional Neural Network (CNN) is a type of neural network made up of several convolutional layers which apply a sliding filter over the input reducing its dimension and obtaining its features. Typically, these layers are followed by one or more fully connected layers that perform the prediction using the mentioned features. This architecture makes CNNs an excellent choice to deal with data in a matrix structure such as an image because this input is too massive for manual feature engineering.

The problem reviewed in this paper can be represented as a Classification problem since it is possible to use a sequence of images that must be classified as a particular future action class. Using Fig. 5 as an example, the high-five action should be predicted before the frames that contain it are captured. The previous work with this kind of algorithm mainly includes convolutional and recurrent neural networks, with the latter being the most common.

Fig. 5. Action Anticipation using Supervised Learning diagram[7]

Since most work uses images as input, transfer learning is also common in the literature. This type of learning involves using a neural network, usually convolutional, that was pre-trained in another dataset. Depending on the goal of its use, these networks can be used entirely or partially; optionally, they can also be trained partially or fully. Some of the most popular examples include VGG-16 and ResNet-50.

In [3], the authors aimed to predict the following action using a Long Short-Term Memory (LSTM) neural network, one of the most common RNNs. In their work, they used a dataset captured with an RGB camera. From these images, they obtained the objects in the environment, the human skeleton joints extracted over time using OpenPose, and the gaze derived from the joints. Then the three data sources were given to the LSTM as input to perform the desired classification. In this process, the authors use an adaptive threshold on the uncertainty of the recurrent neural network, which makes the model need a certain level of certainty to classify the action as a particular class. This creates a more robust solution since a standard supervised learning algorithm would predict the class with the highest probability even if the model has low certainty about every category.

In [10], the authors aimed to predict the subsequent actions that someone wearing a camera would perform and the objects he would interact with. They used three datasets containing RBG frames from which they derived the optical flow and the things in the environment. This data is then passed on to a Rolling-Unrolling LSTM. The Rolling LSTM (R-LSTM) is a network that continuously encodes the received observations and keeps an updated summary of the past. When it is time to make predictions about future actions, the Unrolling LSTM (U-LSTM) is used with its hidden and cell states equal to the current ones of the R-LSTM.

In [15], the authors used an encoder-decoder recurrent neural network topology to predict human actions and intent where the encoder and the decoder are both LSTM cells. At each step, the decoder returns a discrete distribution of the possible actions making this algorithm able to consider multiple action sequences, which are then subject to a pruning method that reduces them to obtain the right action finally. In their work, these algorithms were tested in two different datasets, one containing RGB images with optical markers and gaze information from wearable sensors and another with RGB-D images.

In [18], the authors aimed to predict the intention of the human worker to provide him with the required piece. To achieve this, they used an RGB camera to capture the data from the environment. Then the images are given to a convLSTM framework where the CNN part is in charge of extracting features from the input images, and these features are then passed on to the LSTM to predict the intention. Additionally, another CNN is in charge of recognizing the required piece when the robot is fetching it. This article also tackles the issue of having several possible assembly orders. It solves it by creating a phase at the beginning of the collaboration in which the robot learns the assembly actions and their order from a demonstration.

In [6], the authors aimed to increase the natural collaboration between the robot and the human in an assembly station by interpreting implicit communication cues. The data related to the environment was captured using an RGB-D camera. This data was then passed on to a ResNet-34, a pre-trained neural network that extracted the features from the images. These features are used as the input to an LSTM to perform human action recognition.

In [7], the authors aimed to predict future frames while at the same time predicting the following action. In their implementation, they used public datasets with videos from which they obtained RGB images and optical flow streams.

To consider both sources of data, they also used two ResNet-50's, which are pre-trained networks, one to get the input features from the image and another from the optical flow, and 2 LSTMs to take into account both sequences of inputs. Then the two results are merged into a final classification. They also used two Generative Adversarial Networks (GAN) to generate the subsequent frames, but this is different from the focus of the analysis.

In [19], the authors used video datasets to train a model that would predict a future action from the observed frames. They used three pre-trained neural networks in their work: VGG-16, TS, and ConvNet, to extract features from the images. Then these features were aggregated using a Temporal Transformer module (TTM), and finally, a progressive prediction module (PPM) would anticipate the worker's future action. This article also addresses the issue of specifying what the algorithm should consider as an action. Although most of the literature often implies that the last frames captured by the camera are considered an action, given that those are the frames that contain the last action made by the user, the authors of this article go into greater detail. They tested and evaluated how many frames should be considered as the last action to obtain the best results using a metric from [20] named per-frame calibrated average precision (cAP) calculated with (1). In [19] it is defined with

$$cAP = \frac{\sum_k cPrec(k) * I(k)}{P}, \tag{1}$$

"where calibrated precision $cPrec = \frac{TP}{TP+FP/w}$, $I(k)$ is an indicator function that is equal to 1 if the cut-off frame k is a true positive, P denotes the total number of true positives, and w is the ratio between negative and positive frames. The mean cAP over all classes is reported for final performance.".

In [9], the authors aimed to predict the following action by first predicting the following motion images. They used datasets containing videos and then processed them to obtain motion images. These motion images become the input of a convolutional autoencoder network that generates the following motion images. These images are then passed to a Convolutional Neural Network (CNN) that processes them and makes action predictions for the future. The final action prediction is obtained from the results of the previous network and those of a second CNN, which analyzes the original RGB images.

In [8], the author's goal was to predict the following action someone wearing a camera would perform after some time. Initially, the optical flow was obtained from the captured images, and both were used as input to the model. The model is comprised of a Temporal Segment Networks (TSN), a CNN, and an LSTM to predict the future frame features and then use them to perform the required classification.

Apart from deep learning, there are also more classical approaches such as [4], where the authors aimed to reduce the delay in the robot's response by predicting the human worker and providing a screw or a plate accordingly. They captured the environment using an RGB camera and tracked the hand using optical markers. Then they predicted the following human action using a look-up table containing different orders for assembly actions. With the nearest neighbor algorithm, the actions of the human would be matched with a particular order. If the robot eventually notices that it did the wrong action, it would then follow a hard-coded contingency trajectory to return to the pre-grasping position. The limitation of this method is that all possible sequences need to be on the table because if they are not there, then the robot will match with a different order which may be undesirable.

B. Reinforcement Learning

In Reinforcement Learning, the model is trained to decide which action to take in a specific environment to maximize a particular reward function. These algorithms learn through trial and error using the reward they obtain in each iteration to improve their performance continuously. This type of learning has a certain resemblance to how humans gain knowledge, and it is useful when there is a need for an agent to make decisions in an environment that has considerable complexity, such as controlling a robot or playing a game.

In [21], the authors aimed to make the human-robot interaction more natural by detecting unexpected conditions where the human will not need the robot's assistance, such as when the human's current intention is unknown or irrelevant to the robot or when even though the human's intent is relevant, that task is done only by the human. They used the algorithm Partially Observable Markov Decision Process (POMDP) to achieve this. The training was done with simulation with the model learning a policy by having a positive reward if the task was accomplished and a negative reward if the robot tried to help the human in a situation where it should not.

IV. Human-Robot Collaboration Safety

Finally, safety is a topic that must always be mentioned when robots work with humans, especially in human-robot collaboration. Although collaborative robots or cobots nowadays are made so that if they are interrupted in their work, they switch to a safety mode, effectively stopping, they are still machines with significant strength and can potentially harm the user.

Firstly, just anticipating the actions of the human worker is already a safety measure since it increases the robot's ability to avoid collisions.

In [18], the authors defined speed limits on the robot and ensured that the robot would avoid the workspace of the human. Then when it needs to move closer to the user, its speed is reduced to guarantee the user's safety.

In [22], the authors used deep deterministic policy gradient (DDPG) to plan the robot's trajectory so that the robot would not collide with the human to guarantee his safety.

In [23], the authors attempted to create a sense of anticipation in humans towards the robot's movements through visual cues of the robot's upcoming action, which is the reverse of what it is being tried to achieve in the other reviewed papers. As with the previous article, they also made it so the robot

must reduce its movement speed when close to the robot. Although it was only tested in the Virtual Reality simulation shown in Fig. 6, where the users feel safer, they concluded that the efficiency of the collaboration was increased, and the user had a greater feeling of safety and trust. Furthermore, knowing what the robot will do next also decreases the risk of a collision since the user will avoid the space where the robot is working, increasing safety.

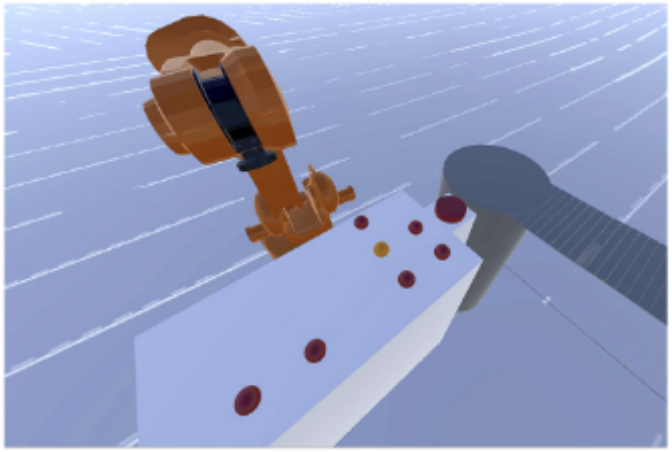

Fig. 6. VR Simulation of [23], the orange means the robot will pick that puck next

In [5], it is also referred that limiting the power and force of the robot decreases the gravity of the consequences of a possible collision, increasing safety. Nowadays, this feature is already included in collaborative robots.

V. Conclusion

In this paper, a review of several articles about human-robot collaboration was conducted, and, more particularly, action anticipation.

Regarding sensors and data sources, RGB cameras are predominant, although there is work with others, such as RGB-D cameras or wearables, to a lesser extent.

Regarding algorithms, there was also a particular focus on machine learning, given that most work nowadays takes advantage of the progress made in that field. Supervised learning was the technique that received a greater focus since it is the most adequate to solve the problem in this paper. With the continuous evolution of machine learning, it is expected that the algorithms related to the topic in this paper also evolve and, consequently, give rise to even better solutions.

Regarding safety, this is a topic common to action anticipation since it is relevant to human-robot collaboration in general. However, as it was seen, anticipating the worker's action can help with his safety.

Despite the work already done on this topic, this is still a relatively new idea, with most of the articles being very recent and the oldest reviewed being from 2016.

VI. References

[1] WiredWorkers, *Cobots*, Last accessed 3 January 2023. [Online]. Available: https://wiredworkers.io/cobot/.

[2] R. Rosen, *Anticipatory Systems: Philosophical, Mathematical and Methodological Foundations*. Elsevier, 1985, pp. 339–347, ISBN: 9780080311586. DOI: 10.1016/C2009-0-07769-1.

[3] C. Canuto, P. Moreno, J. Samatelo, R. Vassallo, and J. Santos-Victor, "Action anticipation for collaborative environments: The impact of contextual information and uncertainty-based prediction," *Neurocomputing*, vol. 444, pp. 301–318, Jul. 2021, ISSN: 09252312. DOI: 10.1016/j.neucom.2020.07.135. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0925231220317719.

[4] G. J. Maeda, A. Maloo, M. Ewerton, R. Lioutikov, and J. Peters, "Anticipative interaction primitives for human-robot collaboration," 2016, ISBN: 9781577357759.

[5] D. Mukherjee, K. Gupta, L. H. Chang, and H. Najjaran, "A survey of robot learning strategies for human-robot collaboration in industrial settings," *Robotics and Computer-Integrated Manufacturing*, vol. 73, p. 102 231, Feb. 2022, ISSN: 07365845. DOI: 10.1016/j.rcim.2021.102231. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0736584521001137.

[6] D. Moutinho, L. F. Rocha, C. M. Costa, L. F. Teixeira, and G. Veiga, "Deep learning-based human action recognition to leverage context awareness in collaborative assembly," *Robotics and Computer-Integrated Manufacturing*, vol. 80, p. 102 449, Apr. 2023, ISSN: 07365845. DOI: 10.1016/j.rcim.2022.102449. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0736584522001314.

[7] H. Gammulle, S. Denman, S. Sridharan, and C. Fookes, "Predicting the future: A jointly learnt model for action anticipation," IEEE, Oct. 2019, pp. 5561–5570, ISBN: 978-1-7281-4803-8. DOI: 10.1109/ICCV.2019.00566. [Online]. Available: https://ieeexplore.ieee.org/document/9009844/.

[8] Y. Wu, L. Zhu, X. Wang, Y. Yang, and F. Wu, "Learning to anticipate egocentric actions by imagination," *IEEE Transactions on Image Processing*, vol. 30, pp. 1143–1152, 2021, ISSN: 1057-7149. DOI: 10.1109/TIP.2020.3040521. [Online]. Available: https://ieeexplore.ieee.org/document/9280353/.

[9] C. Rodriguez, B. Fernando, and H. Li, *Action Anticipation by Predicting Future Dynamic Images*. 2019, pp. 89–105, ISBN: 9783030110147. DOI: 10.1007/978-3-030-11015-4_10. [Online]. Available: http://link.springer.com/10.1007/978-3-030-11015-4_10.

[10] A. Furnari and G. M. Farinella, "Rolling-unrolling lstms for action anticipation from first-person video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, pp. 4021–4036, 11 Nov. 2021, ISSN: 0162-8828. DOI: 10.1109/TPAMI.2020.2992889. [Online]. Available: https://ieeexplore.ieee.org/document/9088213/.

[11] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," Nov. 2016. [Online]. Available: http://arxiv.org/abs/1611.08050.

[12] T. Simon, H. Joo, I. Matthews, and Y. Sheikh, "Hand keypoint detection in single images using multiview bootstrapping," Apr. 2017. [Online]. Available: http://arxiv.org/abs/1704.07809.

[13] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields," Dec. 2018. [Online]. Available: http://arxiv.org/abs/1812.08008.

[14] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," Jan. 2016. [Online]. Available: http://arxiv.org/abs/1602.00134.

[15] P. Schydlo, M. Rakovic, L. Jamone, and J. Santos-Victor, "Anticipation in human-robot cooperation: A recurrent neural network approach for multiple action sequences prediction," IEEE, May 2018, pp. 1–6,

ISBN: 978-1-5386-3081-5. DOI: 10.1109/ICRA.2018.8460924. [Online]. Available: https://ieeexplore.ieee.org/document/8460924/.

[16] F. Semeraro, A. Griffiths, and A. Cangelosi, "Human–robot collaboration and machine learning: A systematic review of recent research," *Robotics and Computer-Integrated Manufacturing*, vol. 79, p. 102432, Feb. 2023, ISSN: 07365845. DOI: 10.1016/j.rcim.2022.102432. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0736584522001156.

[17] K. Kato, W. H. Chin, Y. Toda, and N. Kubota, "A multi-channel episodic memory model for human action learning and recognition," IEEE, Oct. 2018, pp. 843–849, ISBN: 978-1-5386-6650-0. DOI: 10.1109/SMC.2018.00151. [Online]. Available: https://ieeexplore.ieee.org/document/8616147/.

[18] Z. Zhang, G. Peng, W. Wang, Y. Chen, Y. Jia, and S. Liu, "Prediction-based human-robot collaboration in assembly tasks using a learning from demonstration model," *Sensors*, vol. 22, p. 4279, 11 Jun. 2022, ISSN: 1424-8220. DOI: 10.3390/s22114279. [Online]. Available: https://www.mdpi.com/1424-8220/22/11/4279.

[19] W. Wang, X. Peng, Y. Su, Y. Qiao, and J. Cheng, "Ttpp: Temporal transformer with progressive prediction for efficient action anticipation," *Neurocomputing*, vol. 438, pp. 270–279, May 2021, ISSN: 09252312. DOI: 10.1016/j.neucom.2021.01.087. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0925231221001697.

[20] R. D. Geest, E. Gavves, A. Ghodrati, Z. Li, C. Snoek, and T. Tuytelaars, "Online action detection," Apr. 2016. [Online]. Available: http://arxiv.org/abs/1604.06506.

[21] O. C. Görür, B. Rosman, F. Sivrikaya, and S. Albayrak, "Social cobots," ACM, Feb. 2018, pp. 398–406, ISBN: 9781450349536. DOI: 10.1145/3171221.3171256. [Online]. Available: https://dl.acm.org/doi/10.1145/3171221.3171256.

[22] X. Wu, L. Yi, M. Klar, M. Hussong, M. Glatt, and J. C. Aurich, *Intelligent Robotic Arm Path Planning (IRAP2) Framework to Improve Work Safety in Human-Robot Collaboration (HRC) Workspace Using Deep Deterministic Policy Gradient (DDPG) Algorithm*. 2023, pp. 179–187, ISBN: 9783031183256. DOI: 10.1007/978-3-031-18326-3_18. [Online]. Available: https://link.springer.com/10.1007/978-3-031-18326-3_18.

[23] L. Psarakis, D. Nathanael, and N. Marmaras, "Fostering short-term human anticipatory behavior in human-robot collaboration," *International Journal of Industrial Ergonomics*, vol. 87, p. 103241, Jan. 2022, ISSN: 01698141. DOI: 10.1016/j.ergon.2021.103241. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0169814121001591.