

Robot Workspace Monitoring using a Blockchain-based 3D Vision Approach

Vasco Lopes Nuno Pereira Luís A. Alexandre

Universidade da Beira Interior, Instituto de Telecomunicações
Rua Marquês d'Ávila e Bolama, 6201-001, Covilhã, Portugal

{vasco.lopes, nuno.pereira, luis.alexandre}@ubi.pt

Abstract

Blockchain has been used extensively for financial purposes, but this technology can also be beneficial in other contexts where multi-party cooperation, security and decentralization of the data is essential. Properties such as immutability, accessibility and non-repudiation and the existence of smart-contracts make blockchain technology very interesting in robotic contexts that require event registration or integration with Artificial Intelligence. In this paper, we propose a system that leverages blockchain as a ledger to register events and information to be processed by Oracles and uses smart-contracts to control robots by adjusting their velocity, or stopping them, if a person enters the robot working space without permission. We show how blockchain can be used in computer vision problems by interacting with multiple external parties, Oracles, that perform image analysis and how it is possible to use multiple smart-contracts for different tasks. The method proposed is shown in a scenario representing a factory environment, but since it is modular, it can be easily adapted and extended for other contexts, allowing for simple integration and maintenance.

1. Introduction

Blockchain is mostly used for monetary purposes, serving the objective of allowing and registering transactions containing tokens, an idea which has been initially proposed to be used as the basis for Bitcoin [26]. This has gained increasing visibility and wide-spread usage due to inherent properties such as non-repudiation, replication and decentralization of the data, irreversibility, accessibility, (pseudo) anonymity and because it can achieve consensus within an untrusted network [42]. Consensus algorithms have been massively studied, both to solve security problems and to improve the speed that it takes for peers to validate blocks, and the most common ones being used are Proof-of-Work,

that requires miners to solve a cryptographic puzzle to validate a block, which is being used in Bitcoin and Ethereum [37] and Proof-of-Stake, which has many implementations to tackle different problems, but ultimately uses the amount of tokens to select a miner to validate a block [39]. This technology has been used in applications outside the financial scope, mostly due to the fact that blockchain is a powerful ledger that stores data in an immutable way and because smart-contract technology allows it to have actions upon the network without requiring human intervention. One of the most studied applications of blockchain outside the financial scope is in healthcare, where it has been used to privately store medical data with guarantees that only allowed people can see the registries [41], and to improve and accelerate healthcare research [21]. Blockchain has also been used in transportation systems [40], and extensively studied to integrate Internet-of-Thing (IoT) devices [6, 15]. From the proposed methods that integrate blockchain with multiple fields, one of the most promising is the integration with Artificial Intelligence (AI). Both blockchain and AI are disruptive technologies that have immense utility and that have shown they can solve real problems by themselves. The integration of these two can solve problems that are inherent to blockchain [22], such as sustainability and efficiency by improving the energy consumption [5] and by providing dedicated mining, *e.g.* for neural network training [4, 27]. AI can also be used to improve blockchain security [31] and smart-contracts, by aiding in formal verification of the code, which leads to fewer developer induced errors [23]. There are proposals that integrate the two in the form of decentralized applications that use AI for processing data and blockchain as a ledger. However, most of the proposals consist in using blockchain as a lower-level ledger to store monetary transactions and an upper-level application that contains a marketplace for selling AI algorithms and datasets. SingularityNET [33] is an example of this, in which a decentralized market of AI algorithms is coupled to an application that subcontracts other algorithms to accomplish a determined task. Using AI with blockchain opens the possi-

bility of integrating robotics with it. There have been some initial proposals that try to integrate this to enhance individual robot characteristics and to conceptualize how a swarm of robots can be controlled over a blockchain and the benefits of such a system [17]. Work conducted by E. C. Ferrer [3] presents the benefits of this integration in a robotic swarm, showing that it presents the benefits of having global information and allowing for a faster behaviour change of the swarm. In [10], a conceptualisation of a method that allows the sharing of critical data among robots by using a blockchain as a ledger is proposed, introducing the possibility to tackle privacy issues regarding the use of personal data by robots during a Human-Robot Interaction. T. L. Basegio *et al.* [2] allocate tasks in a multi-agent system by having an application that requires Human interaction to define new tasks to be done and stores that information inside a blockchain. In [36], the authors built knowledge processors that use the information stored on blockchain to create coalitions of robots. In [18], a method for detecting anomalies in robots by using blockchain is proposed. In this, the authors use the blockchain to store events of the robots and smart-contracts to detect when anomalies occur. V. Strobel *et al.* [35, 34] show how a reputation system in a blockchain can handle byzantine robots in order to achieve consensus in such networks. Despite these proposals indicating that the integration of robotics and blockchain is achievable and useful, this integration is still in its infancy since blockchain is only used either to achieve consensus on a swarm or to store information. In collaborative robotics contexts, vision can be important to have and there is a lack of proposals focused on integrating vision applications with blockchain. The most prominent work is the one done by ABBC foundation, who proposed a method to improve the security of wallets that hold tokens by using face recognition to identify the owner of an account [1], and the one proposed by V. Lopes *et al.* [19], that uses information from captured images to control the velocity of robots depending on the quantity of raw material needed to be transported, and blockchain to store the information processed by the Oracles and the robot logs, which is then used by smart-contracts that define the changes to be made on the robots. The Oracles are external parties that interact with the blockchain via smart-contracts.

The integration of robotics, blockchain and complex algorithms, such as the ones used in computer vision, is not trivial because blockchains are usually designed for financial applications or to handle transactions of monetary assets and are not fit to work with such components. In this paper, we propose a method that by using vision, can detect the presence of multiple people inside a robot workspace and have different actions on the robots depending if the workspace has entered by known or unknown people. The method uses a blockchain to store the logs of the Robot, which ensures that all the actions that were taken by a robot

are stored and can't be tampered with, the identifications of the images taken by the cameras and the analytics done by the Oracles performing image analysis. We can do this work by using multiple smart-contracts to serve both as storage and to define the changes to be performed on the robots. The method proposed can be adjusted to different needs since the algorithms that run on the Oracles can be any and new smart-contracts for new needs can easily be added. Even though people detection, tracking and identification are well studied subjects [38, 28, 16, 29], the important focus of our proposal goes beyond the algorithms used to perform the tasks described, to focus on demonstrating how it is possible to integrate robotics with blockchain and how to conduct complex image analysis to change the state of robots using smart-contracts, while keeping immutable registries.

The main contributions of this work can be summarized as follows:

- We show how it is possible to conciliate blockchain with computer vision (AI);
- We present a real case scenario where detection of people inside a Collaborative Robot (Cobot) workspace is achieved, in such a way that it is possible to register not only who enters the robot workspace, but also:
 - Unalterably keep the identity of the person (including timestamp);
 - Control the robot in order for it to adapt its behaviour depending on the identity of the person (if she/he is known or unknown).
- We propose a method that can be used to increase security in environments where humans and robots work closely and that can be easily altered, maintained and adapted to more robots, environments and tasks.

The remainder of this paper is organized as follows: First, we explain the proposed method and, the different components that are part of it. Then, we present the experiences conducted. After, we discuss the proposed method and the results of the experiences and finally, we provide a conclusion.

2. Proposed Method

2.1. System Description

The method proposed in this paper consists of controlling robots to avoid collisions with humans by blockchain and 3D vision. Even though Cobots are normally equipped with sensors to stop upon collision, it is important to stop robots before a collisions happens, ensuring that there is no damage neither to the robot nor to the Human that violated the robot workspace. To achieve this, we use RobotChain [8] as a decentralized ledger to store robotic events and

other information in a secure and fast way. RobotChain is a consortium blockchain designed for factory environments, capable of handling a large amount of transactions per second, that implements off-chain protocols to handle the exponential growth in size that blockchains have [9]. We used oracles to process images and smart-contracts to handle the storing of the information and the logic to control robots. This is a generalist approach and can be used to control any robot as long as the robot allows external commands to adjust its velocity and to stop it. This method is also useful as it does not require Human intervention to control the robots, since the smart-contracts automatically perform action-triggers when the premises are met, and it is easy to maintain and adapt to handle new robots and tasks, since it is only needed to add a new smart-contract to handle the new requirements. To avoid collisions, we use 3D images to detect people using the method proposed in [24, 25] and, if there are people present in the warning or the critical zone, we then use [11] to detect faces and identify who they are. Figure 1 shows a representation of this setup that uses a Kinect to acquire depth information about the scene and defines two zones surrounding the robot, the warning and the critical zone, which are represented in the Figure by yellow and red respectively. In our approach, if someone is known (allowed to be near robots, *e.g.* a factory manager) and enters the warning zone, no changes are made to the normal functioning of the robot, but that event is registered in the blockchain. If a person enters the red zone, the velocity of the robot is reduced to 10 seconds per movement and this event and the event that represents the detection and identification of a person are both registered in the blockchain. When the person is unknown (not allowed to be there, *e.g.* a visitor) and enters the warning zone, the velocity of the robot is reduced to 10 seconds per movement, and if then he/she enters the critical zone, the robot is stopped.

Figure 2 shows the complete architecture of the proposed method. It is possible to see from this that the most important module of this architecture is the blockchain, as it serves as a ledger to store all the information but most importantly, as a way for different modules communicate and to share information about the system. The way the blockchain interacts with the other system components is with smart-contracts. They serve both as storage and to handle different logics, such as determine a robot speed depending on the information from the Oracles. The robots are coupled with a computing unit that handles the interaction with the blockchain and, if needed, with an external database that stores images. This ensures that the blockchain does not grow in size exponentially and, by saving the hash of the images on the blockchain, it ensures that if the images are altered, this will be known. The computing units receive information from the smart-contracts about the changes to be conducted in the robots. In the proposed



Figure 1. Robot workspace monitoring using a Kinect (top-left). Yellow represents the warning zone, red represents the critical zone.

method we have three Oracles, one to handle the task that the Robot used in the experiments is doing, one to detect people based on the 3D image acquired by a Kinect and one to detect faces and identify to whom they belong. The first Oracle is based on the method proposed in [19], in which a robotic arm is performing a pick and place task and uses a camera to detect if there are any raw materials to pick. The second and the third one are used to compute vision algorithms and they share information by using a smart-contract that stores information about people being detected on the 3D images and the identification for those images (which are stored in a database). The third Oracle receives the information that the second Oracle inserted in the blockchain and retrieves the correspondent images to perform its task, which is detecting faces and identifying them. After concluding its work, it inserts information about how many people were detected and their identities (if known) in the blockchain. With this, a smart-contract automatically determines if there are any changes to be conducted on the robot. In figure 3 we show a simplified flow of the method, for a known and an unknown person. Step *a* represents the detection of a person in the robot workspace, *b* represents the face detection, *c* is the feature extraction and *d* is the identification of the person. If the person is not known, the method returns "Unknown". After these steps, the smart-contract defines the actions to be taken.

The proposed method is a novel approach on how to

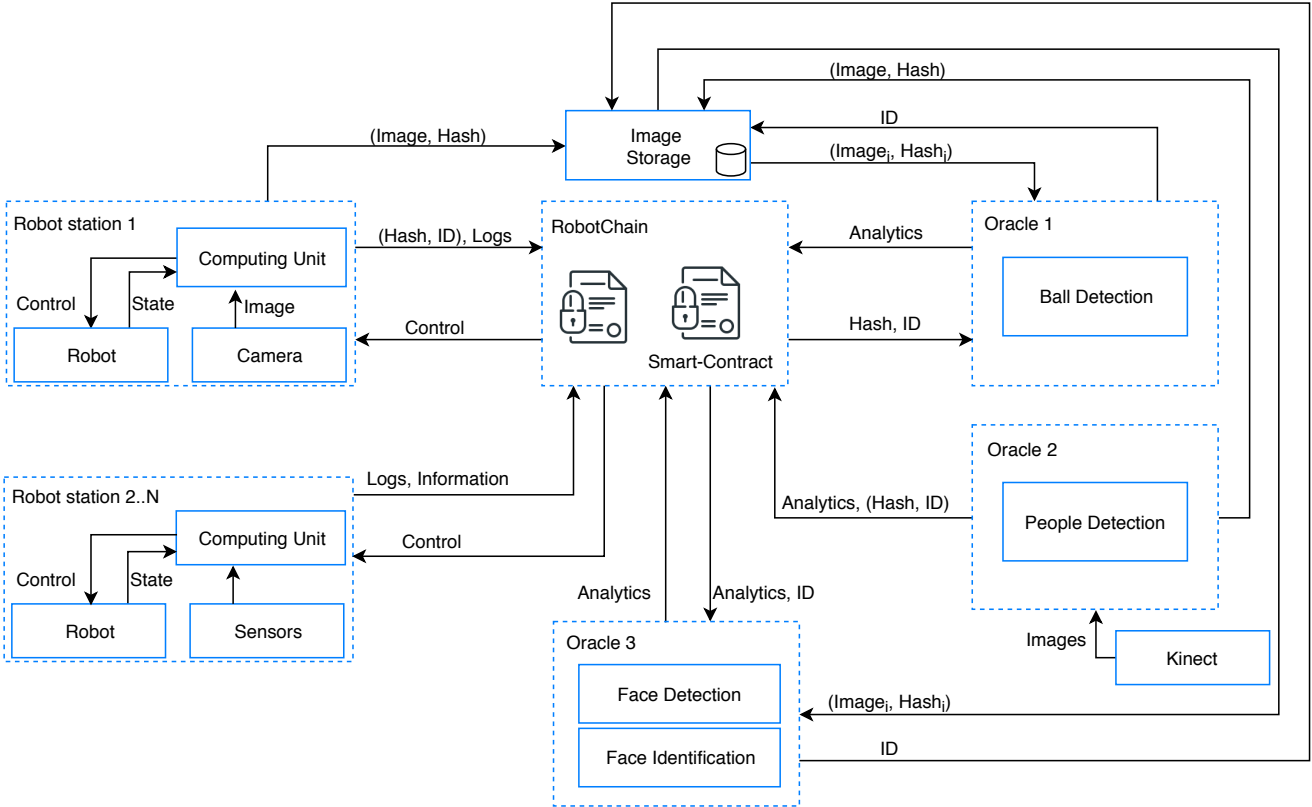


Figure 2. Architecture of the proposed method.

safely validate a robotic workspace and control the robot to adjust to different scenarios. The created method integrates blockchain, Robotics and AI in the form of image processing algorithms, which is an innovative way of dealing with computer vision algorithms and Robotics. By using multiple Oracles to process data we can distribute and decentralize the computation and securely store the information inside the blockchain.

In the following subsections, we explain the components of the system and how they work.

2.2. RobotChain and Oracles

We use RobotChain as a blockchain [8]. RobotChain allows the registering of robotic and other events in a trusted and secure way. It is based on Tezos blockchain [20] that has properties that are essential in robotic environments, such as: a low energy consumption and faster consensus algorithm (Delegated Proof-of-Stake), when compared to traditional ones, *e.g.* Proof-of-Work, its self-amending property that allows changes in the blockchain core to be conducted without the need for hard-forks and the support to formally verify the smart-contracts. This is essential in critical software to ensure that the amount of developer induced bugs is reduced. These can be virtually anywhere, *e.g.* in the

cloud, requiring only a network to communicate with the blockchain network. It is essential that the address of the Oracle(s) is defined on the smart-contract logic. This enforces that only allowed parties can update the information on the blockchain. In the proposed method, there are three separated Oracles performing computation on data (from the blockchain and external sensors, such as a Kinect) and inserting analytics on smart-contracts that are dedicated to: control the robot, store information about people that enter the robot workspace and about their identities. The smart-contracts allow the system either to be private and closed (not allowing external parties to communicate) or to allow trusted parties to interact with it, opening the opportunity for complex algorithms to be performed and to allow other applications to work over them, such as monitoring applications and others.

2.3. People Detection

To detect people we use 3D images captured by a Kinect and process them with the Point Cloud Library [30]. We based our approach in the one proposed in [24, 25]. The method runs on CPU and is capable of detecting people in real-time, even if they are grouped or near walls. This is achieved by first performing a depth-based sub-clustering

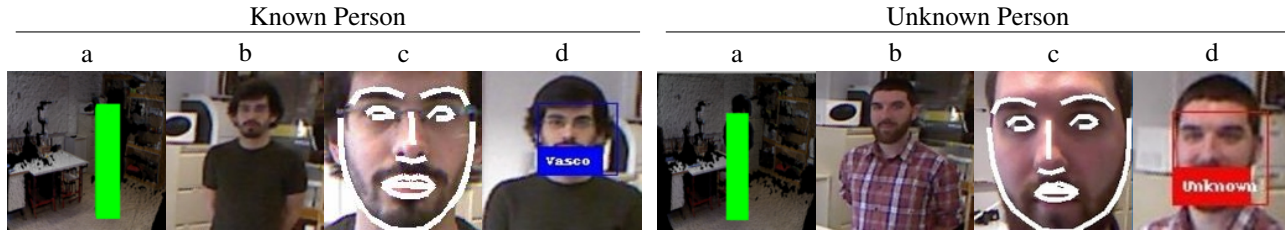


Figure 3. Detection of known and unknown people. Step A represents the people detection algorithm, B the face detection, C the feature extraction and D the identification of the person.

method and then using a Support Vector Machine to detect people in the RGB-D data. In Figure 4 we show an example of two images where on the first one there are no people on the scene and on the second one there is a person detected and a green bounding box is drawn. This bounding box is used to find the position of the person, allowing us to know if the person is inside either one of the restricted zones and if so, use the detection and identification face method only on the region represented by the bounding box. This method is robust as it works in different scenarios since it has the premise that every person starts from the ground. This requires the algorithm to have a ground plane equation to estimate where the ground is. This method is showed to work well when people are standing up or seated on chairs. We changed this method to include the warning and critical areas aforementioned and to extract the image correspondent to the bounding boxes of the detected people in order to send that information to the blockchain.

2.4. Face Detection and Identification

The pipeline of an algorithm that aims to do face recognition contains several steps: first, it needs to find all the faces on a picture; then, for each face, it needs to be able to understand the direction that the face is facing, since it is very rare that a person is looking directly at the camera; third, it needs to be able to extract the unique features of the face, so that those features can be used to distinguish one person from another. The final step is normally the comparison of the unique features of the face with all the people that are already known. This serves to determine if the person is known or unknown to us (identifying to whom the face belongs).

To detect faces and identify them, we based our approach on a library called Face_recognition [11]. This library is a facial recognition API for Python that interfaces with *dlib* [14]. *Dlib* is a toolkit to aid in the development of machine learning algorithms and data analysis tasks and it is widely used due to its efficient runtime. Face_recognition uses different algorithms to tackle the steps aforementioned in a face recognition algorithm. In the first stage, an Histogram of Oriented Gradients (HOG) [7] is used to find faces in an image. On the second stage, it needs to figure out the

pose of the face by finding the main landmarks in the face. These landmarks are found using the face landmark estimation method [13], which is based on an ensemble of regression trees. Once landmarks are found, they are used to warp the image so that the eyes and mouth are centred. After this, the information is then used in FaceNet [32], which is trained to measure 128 features of a face. This Neural Network is used to get simpler data from the image to have a faster search time on the features instead of doing traditional methods such as comparing pixels or sliding windows. In the end, a linear Support Vector Machine classifier is used, to identify a person as unknown or known and get the person identifier in the later case.

We based our approach on this method due to the fact that it is easy to use out-of-the-box, it achieved 99.38% on the Labeled Faces in the Wild benchmark [12] and it is easy to maintain, add and remove people. In figure 5 we show an example of the pipeline of the face recognition and identification that we use. The first step is the information about the image received by the people detection algorithm, the second is the face detection and crop, the third is the feature extraction and the last step is the identification of the person.

2.5. Robot Control

The robot is controlled by smart-contracts. This approach is based on the one proposed in [19]. It uses a blockchain to store robot events and depending on the amount of material needed to be picked, a smart-contract changes the speed of the robot to accommodate the needs of the factory. However, this approach uses only one Oracle. In our approach we use a similar method to control the robots but we use multiple Oracles to perform different types of algorithms to achieve the desired behaviour. In the proposed method, the control of the robot is inside a smart-contract, which means that it can't be changed (without having a specific method that allows so) and performs automatically upon meeting the criteria. The criteria are the detection of people trespassing the robot workspace and their identification. If the trespassing happens in the warning zone, it either slows down the robot if the person is unknown or does nothing, if the person is known and allowed

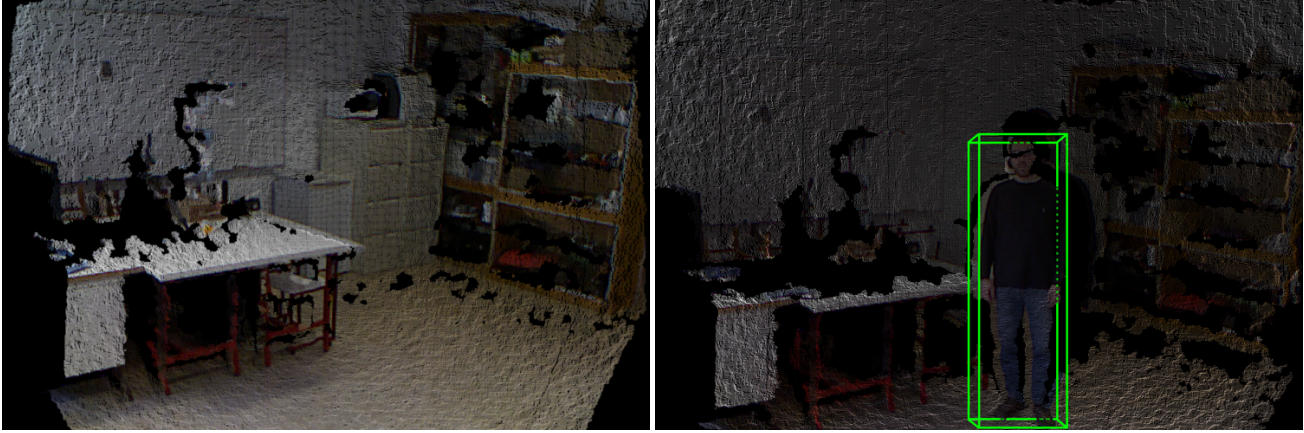


Figure 4. People detection algorithm. The figure on the left was captured when there were no people on it, the second one detected one person and created a bounding box over it.

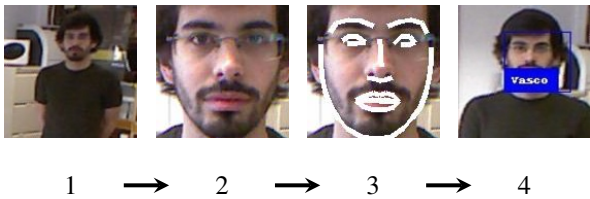


Figure 5. Pipeline of the Face Detection and Identification method.

to be there. In the case that the trespassing happens in the critical zone, if the person is unknown, the robot is stopped, if the person is known, the robot is slowed down. By controlling the robots in such a way, we ensure that they are controlled without Human intervention, that the outcomes of the algorithm are as desired and that every event is registered in a secure and immutable way, meaning that it is possible to see eventual errors and have real-time information about what is and should be happening with the robots.

3. Experiments

To illustrate the working conditions of the proposed method, we conducted two experiments in a scenario where a robotic arm is picking and placing material from one point to another, in which the material returns to the starting point. This simulates a factory job where a robotic arm needs to transport something from one conveyor to another one. Figure 1 represents the scenario created, where the robotic arm picks orange balls from the beginning of the trail and places them at the end, which is slightly higher so that the balls go back to the beginning. The first experiment conducted consisted of a known person entering into the robot workspace, and the second one consisted of an unknown person doing the same. The logic defined in the smart-contract to handle these situations is: if an unknown person enters the warn-

ing zone, the robot must change its speed to 10 seconds per movement, if the person enters the critical zone, the robot must stop. For a known (allowed) person, if he/she enters the warning zone, no changes are performed, if the person enters the critical zone, the robot must change its speed to 10 seconds per movement until the person leaves this zone. The normal speed for the robot is 4 seconds per movement.

In Figure 6, the velocity of the robot in seconds per movement is shown for the experiment with an unknown person. The points A and B represent the moment when the person enters the warning and the critical zone, respectively. In this case, it is possible to see that initially, the robot is working at a constant speed of 4 seconds per movement, once the person is detected inside the warning zone, the smart-contract "fires" the information that the speed should be changed to 10 seconds per movement, which is visible in the change from second 29 to second 30. Then, the person keeps moving inside the warning zone until the second 56, where the person enters the red zone and the robot is stopped. The experiment finished with the robot velocity being 0, which means that it is stopped until new orders. In Figure 7 the same experiment is shown but with a person that is allowed to be near the robot. In this case, there are three points, A that represents the moment a person enters the warning zone, B which represents the entry on the critical zone and C, that represents the moment the person leaves the critical zone. It is possible to see that, because the person is known and allowed to be there, in point A, no changes are made to the robot speed. In point B, as the person enters the critical zone, the velocity of the robot is reduced to 10 seconds per movement (as opposed to stopping if the person were unknown). Then, as the person leaves the area, the robot returns to its normal behaviour, which is a velocity of 4 seconds per movement.

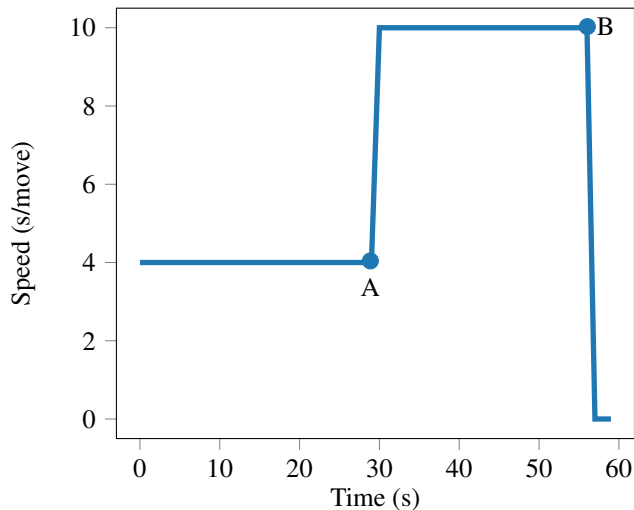


Figure 6. Unknown person experiment. Point A represents the entry in the warning zone, B represents the entry into the critical zone.

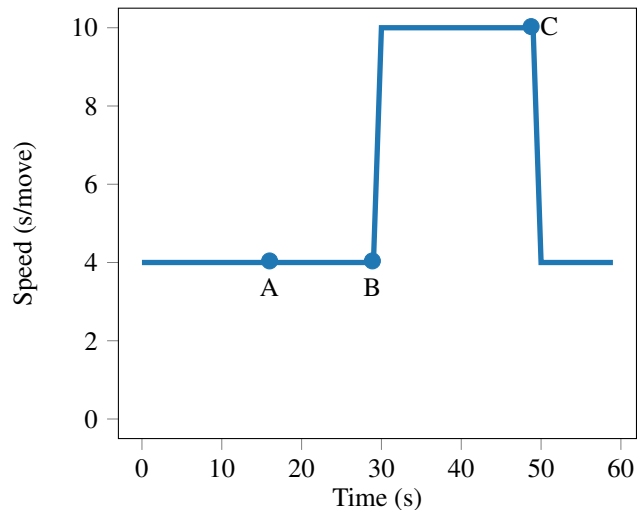


Figure 7. Known person experiment. Point A represents the entry in the warning zone, B represents the entry into the critical zone, C represents the moment when the person leaves the critical zone.

4. Conclusion and Future Work

This paper describes how it is possible to integrate Robotics with vision algorithms while using a blockchain as a decentralized ledger. The proposed architecture is capable of registering robotic events and Oracle information in a secure way and have blockchain to be the gate between all communications and show that this integration of technologies goes beyond a simple way of registering information to more complex behaviours and possibilities. We use Oracles to do heavy computational processing that robots can't do, which leads to more opportunities in the type of tasks the robots can perform. By using a blockchain we also have global information about the whole system in real-time. The proposed architecture is modular, as it is possible to insert new modules that can serve as Oracles and do any type of data processing, or new smart-contracts to handle new tasks and behaviours. This method also ensures that once a smart-contract is deployed to the blockchain, no one can alter its logic and that the whole system can work without Human intervention. The method can be expanded to handle multiple robots just by adding smart-contracts that perform the required actions to communicate with those robots, meaning that a whole swarm of robots can be integrated and they can share information over the blockchain. As a future work, this architecture could be further improved by changing blockchain core features, such as the consensus algorithm to one that does not require any Token, since in robotic environments it is usually not required (when thinking of factories and industrial services).

The development of robotics and AI approaches over blockchain is still in its infancy and will start to flourish

once the blockchain gets more common, easy to use and deploy. In the future, there will be blockchain approaches that are totally focused on integrating multiple parties over the same platform to distribute a way to share information without consensus problems without financial approaches underneath. This methods will be used in the industry 4.0 paradigm and will integrate the Cloud in their inner components, either to work as Oracles or partially store the information of the blockchain.

Acknowledgments

This work was partially supported by the Tezos Foundation through a grant for project Robotchain and partially supported by project 026653 (POCI-01-0247-FEDER-026653) INDTECH 4.0 New technologies for smart manufacturing, cofinanced by the Portugal 2020 Program (PT 2020), Compete 2020 Program and the European Union through the European Regional Development Fund (ERDF).

References

- [1] ABBC. White paper: Abbc. Technical report, March 2018. [2](#)
- [2] Tulio L Basegio, Regio A Michelin, Avelino F Zorzo, and Rafael H Bordini. A decentralised approach to task allocation using blockchain. In *International Workshop on Engineering Multi-Agent Systems*, pages 75–91. Springer, 2017. [2](#)
- [3] Eduardo Castelló Ferrer. The blockchain: A new framework for robotic swarm systems. In Kohei Arai, Rahul Bhatia, and Supriya Kapoor, editors, *Proceedings of the Future Technologies Conference (FTC) 2018*, pages 1037–1058, 2019. [2](#)

- [4] DeepBrain Chain. White paper: Deepbrain chain artificial intelligence computing platform driven by blockchain. Technical report, 2017. 1
- [5] Jianwen Chen, Kai Duan, Rumin Zhang, Liaoyuan Zeng, and Wenyi Wang. An AI based super nodes selection algorithm in blockchain networks. *CoRR*, abs/1808.00216, 2018. 1
- [6] K. Christidis and M. Devetsikiotis. Blockchains and smart contracts for the internet of things. *IEEE Access*, 4:2292–2303, 2016. 1
- [7] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *international Conference on computer vision & Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. IEEE Computer Society, 2005. 5
- [8] M. Fernandes and L. A. Alexandre. Robotchain: Using tezos technology for robot event management. In *Symposium on Blockchain for Robotic Systems, MIT Media Lab*, Cambridge, MA, USA, December 2018. 2, 4
- [9] Miguel Fernandes and Luís A. Alexandre. A Time-Segmented Consortium Blockchain for Robotic Event Registration. *arXiv e-prints*, page arXiv:1904.04306, Apr 2019. 3
- [10] Eduardo Castelló Ferrer, Ognjen Rudovic, Thomas Hardjono, and Alex Pentland. RoboChain: A Secure Data-Sharing Framework for Human-Robot Interaction. *eTELEMED conference*, 2018. 2
- [11] Adam Geitgey. face_recognition the world’s simplest facial recognition api for python and the command line, 2018. 3, 5
- [12] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007. 5
- [13] V. Kazemi and J. Sullivan. One millisecond face alignment with an ensemble of regression trees. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1867–1874, June 2014. 5
- [14] Davis E. King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10:1755–1758, 2009. 5
- [15] Wenjuan Li, Steven Tug, Weizhi Meng, and Yu Wang. Designing collaborative blockchain signature-based intrusion detection in iot environments. *Future Generation Computer Systems*, 96:481 – 489, 2019. 1
- [16] Timm Linder and Kai O Arras. People detection, tracking and visualization using ros on a mobile service robot. In *Robot Operating System (ROS)*, pages 187–213. Springer, 2016. 2
- [17] Vasco Lopes and Luís A. Alexandre. An overview of blockchain integration with robotics and artificial intelligence. *E. Castell Ferrer, T. Hardjono, A. Pentland (Eds.), Proceedings of the First Symposium on Blockchain and Robotics, MIT Media Lab*, Ledger 4.S1 16 (2019) <https://doi.org/10.5915/ledger.2019.171>, December 2018. 2
- [18] V. Lopes and L. A. Alexandre. Detecting robotic anomalies using robotchain. In *2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, April 2019. 2
- [19] Vasco Lopes, Luís A. Alexandre, and Nuno Pereira. Controlling robots using artificial intelligence and a consortium blockchain. *CoRR*, abs/1903.00660, 2019. 2, 3, 5
- [20] L. M’ Goodman. Tezos - white paper. (July 2016):1–17, September 2014. 4
- [21] Polina Mamoshina, Lucy Ojomoko, Yury Yanovich, Alex Ostrovski, Alex Botezatu, Pavel Prikhodko, Eugene Izumchenko, Alexander Aliper, Konstantin Romantsov, Alexander Zhebrak, et al. Converging blockchain and next-generation artificial intelligence technologies to decentralize and accelerate biomedical research and healthcare. *Oncotarget*, 9(5):5665, 2018. 1
- [22] Tshilidzi Marwala and Bo Xing. Blockchain and artificial intelligence. *CoRR*, abs/1802.04451, 2018. 1
- [23] Tshilidzi Marwala and Bo Xing. Blockchain and Artificial Intelligence. *arXiv preprint arXiv:1802.04451*, page 13, 2018. 1
- [24] M. Munaro, F. Basso, and E. Menegatti. Tracking people within groups with rgb-d data. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2101–2107, Oct 2012. 3, 4
- [25] Matteo Munaro and Emanuele Menegatti. Fast rgb-d people tracking for service robots. *Autonomous Robots*, 37(3):227–242, Oct 2014. 3, 4
- [26] Satoshi Nakamoto. Bitcoin: A Peer-to-Peer Electronic Cash System. *bitcoin*, page 9, 2008. 1
- [27] Neuromation. White paper: Neuromation - ”where androids dream of electric sheep”. Technical report, October 2017. 1
- [28] J. Neves and H. Proenca. “A Leopard Cannot Change Its Spots”: Improving face recognition using 3d-based caricatures. *IEEE Transactions on Information Forensics and Security*, 14(1):151–161, Jan 2019. 2
- [29] X. Peng, N. Ratha, and S. Pankanti. Learning face recognition from limited training data using deep neural networks. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 1442–1447, Dec 2016. 2
- [30] Radu Bogdan Rusu and Steve Cousins. 3D is here: Point Cloud Library (PCL). In *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 9–13 2011. 4
- [31] Khaled Salah, M Habib Ur Rehman, Nishara Nizamuddin, and Ala Al-Fuqaha. Blockchain for ai: Review and open research challenges. *IEEE Access*, 7:10127–10149, 2019. 1
- [32] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. *CoRR*, abs/1503.03832, 2015. 5
- [33] SingularityNET. White paper: Singularitynet: A decentralized, open market and inter-network for ais. Technical report, December 2017. 1
- [34] Volker Strobel, Eduardo Castelló Ferrer, and Marco Dorigo. Managing byzantine robots via blockchain technology in a swarm robotics collective decision making scenario. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pages 541–549, 2018. 2
- [35] Volker Strobel and Marco Dorigo. Blockchain technology for robot swarms: A shared knowledge and reputation man-

- agement system for collective estimation. In *Swarm Intelligence: 11th International Conference, ANTS 2018*. Springer, 2018. [2](#)
- [36] Teslya, Nikolay and Smirnov, Alexander. Blockchain-based framework for ontology-oriented robots' coalition formation in cyberphysical systems. *MATEC Web Conf.*, 161 EDP Sciences:03018, 2018. [2](#)
- [37] Gavin Wood. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper*, 151:1–32, 2014. [1](#)
- [38] L. Xia, C. Chen, and J. K. Aggarwal. Human detection using depth information by kinect. In *CVPR 2011 WORKSHOPS*, pages 15–22, June 2011. [2](#)
- [39] Jesse Yli-Huumo, Deokyoon Ko, Sujin Choi, Sooyong Park, and Kari Smolander. Where is current research on blockchain technology? - a systematic review. *PloS one*, 11(10):e0163477, 2016. [1](#)
- [40] Yong Yuan and Fei-Yue Wang. Towards blockchain-based intelligent transportation systems. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pages 2663–2668. IEEE, 2016. [1](#)
- [41] Xiao Yue, Huiju Wang, Dawei Jin, Mingqiang Li, and Wei Jiang. Healthcare data gateways: found healthcare intelligence on blockchain with novel privacy risk control. *Journal of medical systems*, 40(10):218, 2016. [1](#)
- [42] Zibin Zheng, Shaoan Xie, Xiangping Chen, and Huaimin Wang. Blockchain challenges and opportunities: a survey. *IJWGS*, 14:352–375, 2018. [1](#)