A Survey of FPGA Robotics Applications in the Period 2010 – 2019

Dimitris Ziouzios[1], Pavlos Kilintzis[2], Nikolaos Baras[1,*], Minas Dasygenis[1]

[1]*Department of Electrical and Computer Engineering, University of Western Macedonia, Kozani, 50100, Greece*

[2]*Department of Mechanical Engineering, University of Western Macedonia, Kozani, 50100, Greece*

A R T I C L E I N F O	A B S T R A C T
	FPGAs constitute a flexible and increasingly popular controlling solution for robotic applications. Their core advantages regarding high computational performance and software-like flexibility make them suitable controller platforms for robots. These robotic applications include localization / navigation, image processing, industrial or even more complex procedures such as operating on medical or human assistant tasks. This paper provides an overview of the publications regarding different robotic FPGA application fields as well as the most commonly-used robot types used for those applications for the 10-year period of 2010-2019. A short description of each paper reviewed is also included, providing a total view of FPGA technology trends in robotic applications over the last decade.

1. Introduction

Field programmable gate array (FPGA) devices are reconfigurable circuits composed of numerous interconnected logic cells. FPGAs contain an array of programmable logic blocks, and a hierarchy of "reconfigurable interconnects" that allow the blocks to be wired together. They are still used as a high-performance computing platform in multiple applications nowadays, despite their overshadowing by CPUs and GPUs. The general core advantages of FPGAs are that they can be easily connected to most types of circuits due to their protocol-independent serial links and they offer hard real-time computations and highly parallel effective programming with low power consumption, using algorithmic high-level synthesis [1]. FPGAs can be reconfigured and reprogrammed according to the user's requirements, unlike fixed hardware structures that CPUs have. This allows users to utilize them in a vast variety of applications.

Robots today are used for multiple tasks, for both industrial and consumer purposes that require high performance and low power consumption. They can be used for manufacturing or logistics purposes, while offering medical, educational, home, defense and other activities in the services' sector. So, the applications implemented on FPGAs in the robotics sector contain several multi-factorial tasks from multiple domains. Primarily those tasks include vision, sound or touch response systems and object tracing through processing of a high data flow coming from

sensor inputs. Furthermore, increased real-time requirements for controlling robotics' functionalities such as the movement of a robot, make FPGAs to be a more attractive processing solution since they can be reconfigured as many times as it is necessary, even during runtime, while they are utilized, until the required functionality is properly achieved. FPGAs can actually rewire their internal circuitry in order to allow configuration, after the control system is installed onto the field. Moreover, the high-power consumption of robotics applications indicates FPGAs as suitable computational systems in comparison to other solutions. In fact, the advantages of using reconfigurable dynamic systems do not exclusively lie on the increased adaptability to changing conditions, but also on reducing power consumption without affecting the robot's capabilities. Overall, the size and speed of an FPGA is similar to the application-specific integrated circuit (ASIC), but their shorter design cycle and reconfigurability capabilities make them more flexible for use in robotic applications [2].

FPGAs are a mature implementation technology, spanning over 35 years of technology improvements. At the same time, after 70 years of evolution in robotics the controlling complexity of them has reached a level that only high-end processing cores can handle. Investigating the blending between the robotics and the FPGAs in the related literature, we located a gap. In this paper, we aim to answer the research question: Given all the advantages in CPUs and GPUs, is it viable to use FPGAs in robotics applications, and if yes which are the areas that FPGAs excel in the robotics domain and what challenges loom ahead? Towards

---

this, we investigated IEEE listed scientific publications in the 10-year period between 2010 and 2019 in order to get an overall view of the typical application fields of FPGAs in robotic applications. Figure 1 shows the absolute quantity of the total publications regarding journals, conferences, early-access articles and magazines in the IEEE Xplore and ACM digital library depicting the actual publication tendency over the last 27 years, since the library includes publication data since 1993. The total publications count from 1993 to 2019 regarding FPGAs in robotics is 1174 publications, including the aforementioned four types of publications. The publication count for the testing period of 2010-2019 counts 815 publications in total, where 762 are conference papers, 49 journal papers, 2 early-access articles and 2 magazine articles. The average publication number over the 10 year investigation period is almost 80 publications per year. As accrued from Figure 1, from 1993 to 2001 the publication count remains relatively steady with less than 10 publications / year, while from 2002 to 2009 there is a yearly significant increase, reaching the top value of 81 publications in 2009. Since then, a fluctuating count is observed in the testing time period of 2010-2019, with the maximum value accomplished in 2019 with 102 publications. Figure 1 imprints the technological boom in the FPGA capabilities that started in the millennium (2000), allowing wide FPGA usage in robotic applications

This tendency lasted for almost a decade, until 2009 and since then the robotic FPGA applications - according to the publications count - remain relatively steady with fluctuating tendencies. Since the total publications for the tested time period are almost 800, the researchers choose to analyze 20 popular publications for each year, in order to imprint the latest scientific tendencies. The total number of publications tested for the 10-year period is 200.

The paper is structured as follows. First, a brief theoretical analysis of the modern robot types regarding their application fields and interaction mechanism is given, according to the latest literature trends. The description of the three main robot categories as classified in this survey, follows. The six main FPGA application fields, as uncovered in the survey analysis, are then quoted.

No. of Publications

Figure 1: FPGA Robotics-related publications in IEEE Xplore and ACM digital library per year

The conclusion section sums up our findings, as indicated by the survey findings. All reviewed literature is quoted with application field and robot type categorization, a short description of each paper and the year of the publication.

2. Related Work

A shortlist of FPGA-related studies uncovering related application fields, architectures and technology trends has been revealed in the survey. The most detailed view for survey applications for a distinct period of time is included in [1]. 22 distinct technology applications of FPGAs (+ "other" quoted as distinct category) for a 15-year period are documented, investigating 815 publications in the IEEE Xplore and ACM digital library for the aforementioned duration. This survey offers a relatively indicative view of the FPGA technology trends until 2015, however it does not investigate the FPGA architectures designed specifically for robotics applications. Paper [3] highlights the availability of reconfigurable applications as platforms for digital controllers and quotes the most important factors for the technology trend of FPGAs utilization in comparison to other architectures, primarily due to the acceleration and flexibility of FPGAs, reduction of costs and reduced energy consumption. Study [4] quotes a general description of FPGAs, regarding their use as accelerators and simulator platforms, the usage of their integrated memory and their fault tolerance as well as the programming languages and computation methodologies. However, it does not quote applications per year or include a specific distinction of application fields The FPGA virtualization techniques are included in [5], separated into three categories - resource level, node level and multi-node level. The first level investigates whether FPGA's resources are reconfigurable or not. The node level explores the applications needed in both software and hardware terms regarding virtualization at a single FPGA. The existing architectures of how to connect multiple FPGAs are described in the multi-node level. The survey [6] investigates FPGA's routing architectures, dividing them into island-style routing and hierarchical routing. Furthermore, it quotes a brief description of the recent technology applications of FPGAs, distinguishing them into eight distinct categories. In conclusion, there are no specific surveys like this one and for this reason we believe our paper will play a pivotal role as a reference for researchers in the field of implementing robotic applications on FPGAs.

3. Application Fields of FPGAs in Robotics

In our survey, we categorized the papers into 6 major categories, according to the application. In order for an application field to be categorized as a distinct category, at least 5 publications had to be identified in the databases. Some application fields can be mixed, for instance, a robotic controller may use a regular FPGA algorithm and Fuzzy logic architecture or a navigation application can use SoC FPGA image processing systems. Distinct categories as well as mixed ones are referred to the Application Field column of Table 1, where the research overview is depicted. The Robot type column includes the type of robot which this publication's application is designed for. A short description of the publication's application regarding the algorithms used, the software or the type of the FPGA controller chips is included in the next column. The publication year and the reference number are depicted in the last two columns of Table 1.

3.1. Classification of Robots

While there is no strict classification of robots, most robotic applications can be either categorized according to their mechanism of interaction or according to their application field [7]. Figure 2 depicts the robot classification according to their mechanism of interaction [7]:
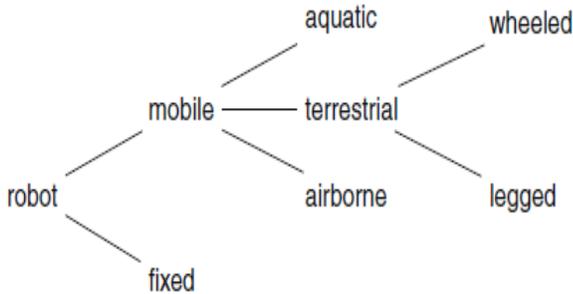

Figure 2: Robot classification by interaction mechanism [7]

The basic classification of robots according to their application field is shown on Figure 3. It has to be noted that the six sub-categories of Fig. 3 can also include variations. For instance, educational robots can either be wheeled or humanoid (legged) and industrial robots can be either articulated robots, cartesian robots, parallel, spherical, or SCARA, etc. [8]. The aforementioned categories are based on the primary means of navigation (for educational robots) or the robot's basic configuration (for industrial robots).

However, this survey includes the robotic applications that are FPGA-applied, so the classification of robots is done according to those categories that can use the application fields found in the survey publications and analyzed in the next chapter. The main robot categories of the FPGA-based robotic applications included in this survey are the following:

- Industrial Robots (A1)
- Autonomous Mobile Robots (A2)
- Humanoid Robots (A3)

For a robot type to be classified as a distinct category, at least 5 publications had to be found for the selected period. In other cases, this column is classified as "other", implying a different type of robot, described in the "description" column such as medical, defense, educational, etc.
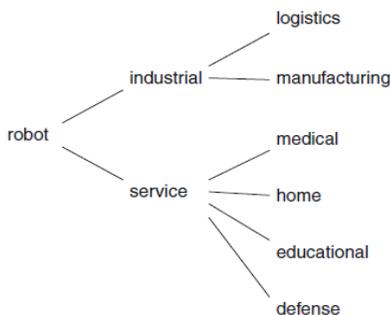

Figure 3: Classification of robots according to their application field [7]

3.1.1. Industrial Robots

Industrial robots constitute one of the major of the FPGA-related publications in this survey, as depicted in Fig. 4. The main subcategories of this main robot category are [10]:

- Articulated robots
- Cartesian robots
- Cylindrical robots
- SCARA robots
- Delta robots
- Polar robots

The articulated robot type is amongst the most common types of industrial robotics. It includes a mechanical arm conjoined to a stable base. Rotary joints connecting the arm associations may vary from two to ten joints although the most common robots are the ones with six joints. The joints give the robot additional freedom to its movement while the joints can be either orthogonal or parallel. Cartesian or Gantry robots are also amongst the most common robot types. They have a rectangular configuration with three prismatic joints (X, Y, Z) allowing linear motion, while some Gantry robots have an additional wrist allowing rotational motion. Cylindrical robots use a main arm making an up-and-down motion. This arm is internally embedded with a cylinder allowing the extension/reaction motion. Cylindrical robots use a 3-dimensional coordinated architecture using a specifically selected reference axis for selecting point positions. Cylindrical robots are mostly used in picking-placing materials operations. SCARA (Selective Compliance Assembly Robot Arm) robots use two parallel joints providing movement at a specifically selected track. The rotary tracks are placed vertically and the operating tooling attached to the arm moves horizontally, making SCARA robots most suitable for vertical assembly operations where efficient and quick pick and place operation is critical. The primary characteristic of the Delta robots is their parallel link configuration consisting of parallel joint connections linked to a common stable base. The positioning of the effector is controlled with the robotic arm making the robot type suitable for speedy pick-and-place applications. Polar (or spherical) robots use stationary arms with spherical work envelopes which can be placed in a polar-coordinated architecture. They use one linear and two rotary joints conjoining the links and a twisting joint conjoined to the arm.

3.1.2. Autonomous Mobile Robots

Autonomous Mobile Robots (AMR) have been classified as a distinct category because they can be used for industrial as well as domestic, medical, defense or educational purposes. An AMR is designed to move on wheels or with legs (human walking simulation is excluded from AMR category) across the ground floor to specified locations operating as a flexible assistant performing numerous tasks. The movement of an AMR is accomplished by using a total of advanced on-board sensors, FPGA-architectures and processors that allow it to comprehend and interpret its external environment. The aforementioned means offer the AMR the capability of combining the information from

multiple different sources such as video or GPS in order to navigate when there is a malfunction or full absence of data from one or more data sources. The core operation of an AMR is its control system, where FPGAs may constitute an implementation solution in many cases. While FPGA-based control systems are analytically explained in the next section, the design requirements of an effective AMR control system typically include:

- **Multiple sensor and processor integration:** As mentioned above, the possible malfunction, unreliability or inapplicability of any individual sensor or set of sensors must be coped with the integration and effective operation of the rest of AMR sensors. Furthermore, AMR must be able to accomplish all assigned tasks despite concurrent and often conflicting actions that need to be performed.

- **Reactivity to environment:** AMR should be capable of reacting to unscheduled sudden changes in the external environment by executing fast and efficient decisions in order to fulfill the assigned tasks. Software and architecture efficiency play a crucial role at that requirement, including the accuracy and usability of FPGA software algorithms.

- **Reprogrammability and Expandability:** An expandable architecture of an AMR allows the incremental build-up of the system and its continuous improvement and evolution. FPGA chips offer a significant advantage over other processing solutions in this case since they are reconfigurable and can be programmed to adopt a different behavior every time is required, changing the configuration of FPGA's internal circuitry.

3.1.3. Humanoid Robots

Humanoid robots are robotic applications that mimic human body structures in both behavioral and motion terms. They are considered a relatively complex robot type since it is necessary to consider their mechanical (musculoskeletal) structure, the computational and sensory nervous system, the applied methods of information processing in the robot's primary decision-making units and the applied software algorithms as a whole and for any given application that they have to perform. FPGA chips can be used for multiple humanoid robotic applications such as their motion control, communication modules between robots and users, visual sensing and processing that constitute the robot's total behavior in mimicking human functionality. However, the primary distinguishing element of humanoids in comparison to other robot types is their human mimetic motion and human-like robot reaction. The first element involves the robot's skeleton structure, the body proportions, the muscle arrangement and the joint performance [10]. Human-like robot reaction mainly represents the robot's responses for safety. This can be applied actively by using a feedback loop or passively by applying a suitable robot's mechanical design. Humanoids with high gains actuators tend to have a low performance on estimating correctly the forces applied by the human for specific functions, while the usage of integrated early-stage additional sensors shows better performance on that specific task [11]. FPGA chips constitute an efficient choice as a fused sensor architecture.

3.2. Application Fields

The sections analyzed below, will briefly introduce the primary application domains of FPGA-related technology components on robotics applications. These applications can be either mathematical and software algorithms, hardware FPGA-oriented architectures, mixed systems with FPGA components or combinations of all the above.

3.2.1. Robotic controllers

A robot constitutes an autonomous closed-loop control application because it is affected by the inputs from its sensors. Due to the highly effective integration of the intellectual properties of the embedded processors in the FPGA, the mathematical algorithms executing the demanding computations required by the robot controllers can be implemented by software in the FPGA. This application field mainly includes the coding of FPGA motion control systems of immobile robotic applications and industrial robots. Industrial immobile robots are widely used in order to solve tasks such as assembly, handling or welding.

Servo control is the most crucial part of robotic controller architecture since a possible failure to maintain a closed-loop behavior may delay the production process, damage the robot itself or even cause accidents and danger for human life. A servo-control solution based on FPGAs is considered amongst the most effective ones. Specifically, an FPGA-based control interface is responsible for multiple tasks in the robotic operation such as decoding feedback position data, processing communication signals and control outputs and transmitting the data to the actuators. The instructions provided to the robot used to perform specific actions are processed by the robotics control system. The programming language is actually used to describe all the variables that affect the robotics operation, primarily regarding movements on specifically set paths. There are several path types that a robot can operate under, depending on its purpose. The simplest type of robot motion is the movement from one point to the other, without having to perform any other parallel simultaneous actions. This motion can be executed as joint displacements and this type of control is called joint control [12]. The joint control motion expresses the interactions of position, acceleration and speed in the robot's language describing the necessary movements to perform the requested action. The main limitation of this control method is that the user needs joint dimensional information since he must comprehend the robot's joint displacements in order to drive the robot into the desired position. This limitation can be overcome by using the Cartesian space, which helps to interpret robot's movements by using Cartesian coordinates, based on the Jacobian Transpose matrix. This type of control is called Cartesian control. Due to this reason, the mathematical algorithms representing the kinematics of Cartesian control motion are significantly important in order to propose the most suitable control structures. The FPGA-based controller is then used to transform the trajectory created in the Cartesian space into a parallel robot's trajectory in joint space. This application field includes research undertaken for kinematics algorithms and software applications regarding the robotic controllers, mainly representing immobile robots executing industrial tasks.

3.2.2. Mapping – Navigation

This application field includes research regarding autonomous mobile robots, analyzing the usage of FPGA controllers in crucial operations, such as robot's localization, mapping and navigation. In general, the navigation of robots is divided into two parts: The 1st part is about the robot being capable of selecting the most suitable path between a starting and a finishing (goal) point. This path can be computed by either making calculations based on pre-loaded existing data or by processing incoming real-time data by GPS and other applications. The 2nd part includes the robot's real-time effectiveness on adapting its behavior to the external environment characteristics, such as avoiding obstacles, making turns and so on. FPGA applications can be used for both parts. Robot navigation is performed by using line following and obstacle avoiding algorithms that can operate with sensor data and probabilistic algorithms that study traffic and external environment patterns. The FPGA control unit is receiving and processing all incoming data from sensors and modules in order to choose the most suitable path.

Localization is another crucial process of mobile robotics as it is used in order to determine a robot's position in relation to an external reference point, called landmark. There are two localization methods: The first method calculates the distance from the landmark and its angle relative to the north, called the azimuth. The second method measures the angles to the object from two different positions, called triangulation. FPGA control units can process data and by applying the proper algorithms are capable of determining the right course. In recent years both localization techniques have been replaced by GPS location and navigation. FPGA receivers are often used to process GPS data and apply the proper navigation algorithms.

Simultaneous Localization and Mapping (SLAM) is the next-in-line step in robot mapping and navigation. A primary task of a mobile robot in an unknown environment is to create and frequently update a map of the environment and compute its current location within this map at the same time. This operation is called the SLAM operation. At normal conditions, the robot can use sensor or odometry measurements to gather information over its environment in order to perform localization and mapping. SLAM is used when these conditions are altered such as the localization at an underwater or space mission. Some researchers have proposed the use of probabilistic methods, such as Extended Kalman Filters [13]. The Extended Kaman Filters (EKF) algorithm simultaneously computes a model map of the environment and the current robot position based on odometric and exteroceptive sensor data. FPGA-based architectures usually implement the EKG algorithm. Alternative solutions avoiding the errors produced by odometry rely on scan-matching techniques that usually rely on range measurements acquired from robot sensors. Such an approach is the SMG algorithm [14]. This algorithm can perform SLAM by applying highly improved parameters which in turn considerably increase the algorithm's computational requirements. A considerable acceleration over software can be managed, by using the programmable resources of FPGA circuits and the parallelization properties of the algorithm.

3.2.3. Image Processing

FPGAs are widely used for the execution of digital image processing algorithms. They are used even more intensively, specifically in the case of real-time embedded applications, where latency and power are considered important issues. Typical topics in digital image processing are the evaluation of single images or numerous frames of a video signal for specifically set criteria such as movement information extraction or object tracking. For example, an FPGA embedded in a camera can execute image processing directly via the transmission of the image through the sensor. The camera provides a fully processed output stream that way, rather than just an image sequence.

The sector of robotics significantly benefits from the advantages provided by digital image processing on FPGAs. The image processing system of a robot is crucial for the enhancement of processing performance, since it includes the ability for spatial recognition and multiple detective algorithms to carry out the mission tasks. These tasks primarily include localization/navigation in the case of autonomous mobile robots. The primary reason for applying algorithms in FPGA robotics is the parallelism option which allows advantageous capabilities such as real-time image filtering for pre-processing purposes. Moreover, the local memory embedded in current FPGA architectures allows the buffering of pertinent image information to limit the communication with external memories, a situation that can potentially form a bottleneck [1].

Furthermore, FPGAs are more effective in comparison to CPUs in several aspects, regarding the ideal image processing system. CPUs execute their operations sequentially, so the first operation must process the entire image before the second operation starts. Since FPGAs are operating in parallel, every operation of the algorithm can operate on different pixels of the image simultaneously. This denotes that the time needed to receive the first processed pixel and process the whole image is much shorter, in comparison to an exclusive CPU application. Even if a combined co-processing FPGA-CPU architecture is used to transfer or receive images from the CPU, the total processing time is still shorter than an architecture that exclusively uses the CPU. Moreover, image processing architectures for robotics based both on USB and PC interface, are usually applied by multithread software solutions, so hard real-time response cannot be achieved for the proper integration of image and behavioral usage. Therefore, a robotic vision system that can perform image processing through an FPGA, which is directly connected to the CMOS sensor is more efficient.

3.2.3. Neural Networks

Artificial Neural Networks (ANNs) can be used in a variety of tasks performed by robots such as trajectory and image recognition, vehicle control, image processing, pattern and sequence recognition and medical diagnostics. They are statistical models destined to adapt and self-program, by applying learning algorithms that make them execute the aforementioned tasks. ANNs operate as parallel distributed computing networks and they operate correspondingly to the biological neural system. According to this, there are multiple input signals directed to neurons. Every input is assigned to a relative weight value which correspondingly influences the impact of that specific input. In

modelling, weight values are the adaptive coefficients within the network that calculate the magnitude of the input signal. The neuron output signal is then created by adding all the weighted inputs algebraically, simulating the biological cell body [15].

In order to perform properly, developers arrange ANN architectures in specific layers that are implemented in parallel. Since ANNs require significant computational power, they are better implemented using parallel hardware architecture [16]. ANNs can be implemented by using either software or hardware implementation. Software implementation is the favorite choice, supporting ANN training and simulation on sequential computers for imitating a large variety of neural network models, with significant flexibility. That is so since the developers do not need to know the internal operations of the neural network elements, but can be exclusively focused on the software application of the ANN. On the contrary, hardware FPGA-based architectures offer high applicability and the significant advantage of parallelism. The two basic characteristics of the FPGA-based reconfigurable computing architecture are its dynamic adoption and parallelism. These architectures are most suitable to apply ANNs as a developer can achieve concurrency and easily reconfigure in order to adjust the topology and the weights of an ANN. In total, FPGA-based architectures offer numerous advantages for neural implementations such as:

- Prototyping through reprogrammability: The neural FPGA-based architectures can be directly tested in order to find the most effective one, without any additional costs. Furthermore, an effective application architecture already developed and implemented can be later replaced by another one, without the need to design a new chip.

- Embedment: FPGA architectures can be used for all kinds of embedded applications, in order to take advantage of the effectiveness of neural calculations.

- On-chip leaning: Usually on-chip learning is avoided as it is considered a complicated process that can easily lead to loss of efficiency due to a number of reasons. On the contrary, off-chip learning is a more preferable choice when there are no requirements for dynamic learning. In the case of FPGAs, on-chip learning can be executed before the chosen implementation of the learned ANN on the same chip.

- Mapping: FPGA-based ANN architectures can be mapped onto more modern and effective FPGAs, whenever technology advancements indicate so. The basic prerequisite for ANN implementation is the scalability level of the implementation method.

3.4. Fuzzy Logic Control

Control algorithms for detailed dynamic models often require the exact specification of accurate target values. These equations may be complex and nonlinear, requiring longer "run-time" in order to get processed. An alternative approach to the usage of classical logic controllers described in section III subsection B.1 is the usage of Fuzzy logic control algorithms, which use imprecise specifications in order to reach the same goal. Fuzzy logic controllers are implemented to applications that are difficult by nature to model precisely. They are often considered more

robust algorithms in planning parameter value changes, in comparison to regular control algorithms, since they do not require high precision and high computational power.

In the sector of robotics, fuzzy logic controllers are often used for automatic navigation purposes. This is due to their capacity to process a high number of imprecise and incomplete input signals. In such a case, signal input processing may allow robots to navigate in uncertain environments. A robot is equipped with multiple sensors that allow it to sense and process the environmental information. Fuzzy controllers receive sensor outputs as inputs for processing according to the fuzzy rule database that dictates the robot's possible behaviors. The robot decision making mechanism implemented by the fuzzy controller consists of three basic steps in order to reach the final decision. First, during fuzzification a crisp input value is converted to a fuzzy value that is performed by the use of the information in the fuzzy rule database. Next, fuzzy inference (rule evaluation) formulates the mapping from a given input to an output using fuzzy logic, applying the rules to the fuzzy inputs. The mapping includes the rule framework under which robot's decision making will take place. Then, during the defuzzification step the values from the output variables are calculated from the fuzzy values resulting in the controller's final decision [17]. In total, fuzzy algorithms allow the robot to choose amongst various combinations of decisions the most effective one, rather than being forced to make a pre-designed single decision.

A fuzzy logic controller is capable of processing inputs from a wide range of sensors. For instance, ultrasonic navigation sensors can calculate the distance between a robot and the obstacles of the external environment. GPS applications can be used for localization purposes, detecting the robot's current position. Fuzzy algorithms constitute a suitable controller solution for navigation in 2D and 3D environments, industrial robot manipulators [18], image processing algorithms [19] and several other robotic domains that require high computational power.

3.2.5. Swarm Robotics

Swarm robotics is the task of using multiple robots to perform different activities simultaneously, rather than using one robotic application with enhanced capabilities. It is considered a distributed approach that aims at coordinating robot behaviour by copying natural decision mechanisms of the biological entities (humans, animals). These algorithmic mechanisms allow a group of robots to achieve performance levels that could never be achieved by a single or a few robotic applications. The usage of multiple robots may significantly shorten manufacturing time, performing different tasks with simpler, scalable and more robust robotic applications. The major issue in swarm robotics is the effectiveness and precision of the coordination algorithms, as they are usually performing on the same physical object or environment.

There are two main approaches of designing swarm robotics. The first approach is the centralized system, according to which a primary robot acts as a leader-coordinator for the rest of the robots of the swarm [20]. In FPGA robotic controllers, this is a relatively simple solution for navigation tasks. In this case, a master (leader) robot plans the path planning to traverse to the direction of a certain goal point, with the rest of the robots to follow. The

behavioral control mechanism creates a constant mutual communication between the leader and the followers, whose behaviour is checked and corrected at specific time periods, according to the leader's motion. The core advantage of this approach is that all follower or slave robots can be simple robotic applications compared to the master robotic unit. The main disadvantage is that if the leader robotic unit fails to perform as planned or properly communicate with the followers, then the whole swarm fails to complete the task.
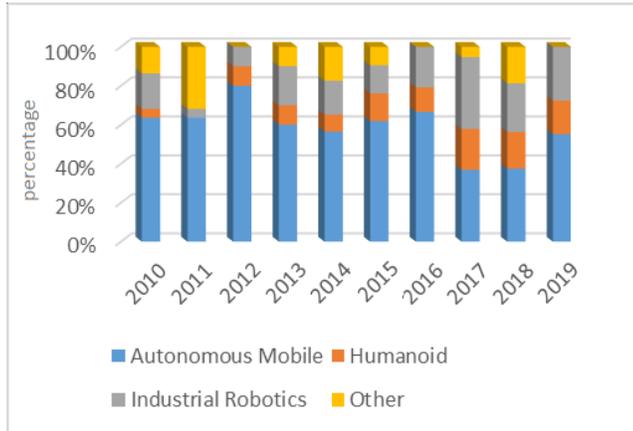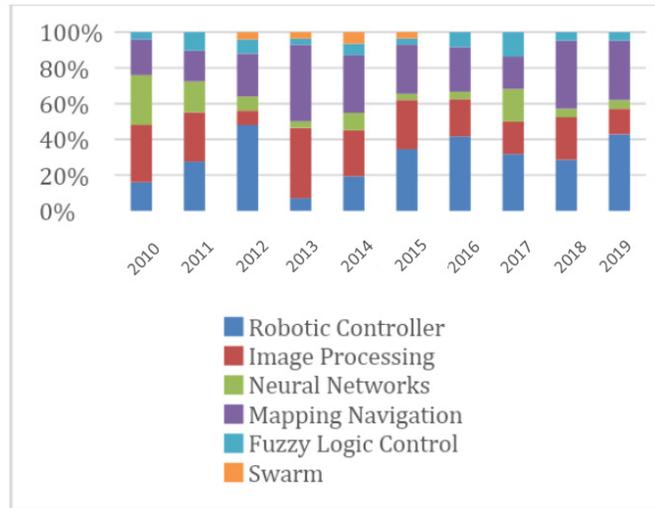

Figure 4: Robot types

Figure 5: Distribution of FPGA applications for the period of 2010-2019

The second approach is the decentralized system that operates without a leader robotic unit and it is not affected by the cases of a single swarm unit failure or miscommunication amongst swarm members (agents). According to this approach, the motion of each robot is coordinated using local information exchange such as speed, obstacle existence or geographical coordinates, by the closest neighbor agents at a given time period. These motions are usually developed according to animal aggregations. Also, there is a physical decentralized system based on the physical interaction among agents. The communication amongst robots is either directional, by exerting forces to each other or in directional by manipulating or placing a common object. Directional communication may be applied by transmitting/receiving digital signals (WiFi, Bluetooth, etc), sound or light. In directional communication may be applied by either leaving optical markings

in the ground or placing/manipulating an object in the environment that will be used as a marking point by other agents [20]. In total, the centralized architectures are simpler and used primarily for navigation purposes while decentralized architectures are more complicated and can be used for performing activities such as simulations, biotechnology or military tasks.

In Figure 4 and Figure 5 the robot types and the distribution of FPGA applications found in this survey are depicted, according to the total count of 200 publications tested.

4. Reviewed Literature

In this section, we present and analyze the examined research publications of the IEEE & ACM database. For each application field, we present the publications by giving a short description for each publication. Only selected publications that utilized high performance, throughput and efficiency were selected. This description for each publication includes a summary of the publication, the key points and its specifications.

4.1. Application Field: Image Processing

In [161] we see an approach about a real time image processing patrol service robot. The robot consists of a CMOS camera module that sends data to the FPGA board. The genetic algorithm optimizes threshold values for the filtering operations according to various lighting environments. The visual navigation system has been developed for the linkage between flexible hardware circuits and real-time software applications for robot vision purposes. Experimental results show the feasibility of the proposed methodology.

In 2013, an FPGA algorithm used to detect corners in colored stereo images by implementing Harris corner detection technique [155]. The FPGA used is the XC5VLX50 and the architecture consists of 3 pipeline modules consisting of OmniVision image sensors in RGB555 format. Data is stored into a single DDR2 memory of Micron. By using less than 75% of FPGA resources and a 100MHz system clock, the authors achieved a corner detection rate of 0.33 pixels per clock cycle (ppcc) corresponding to a corner detection frequency of 54Hz for the stereo images. They highlighted the performance benefits of their methodology by conducting experiments.

A novel human-robot communication algorithm for both voice recognition and image processing embedded in an autonomous mobile robot [149] is proposed. The main controller is built on a Spartan FPGA processor, processing visual and voice information. The result of the research is a very flexible and powerful development tool for human-robot multimodal communication research and implementation. The hardware resources of this robot are wisely integrated by using the LabView graphical software toolkit, enabling a large amount of voice signals and image processing algorithms testing. The designed software drivers also allow various sensor modality including vision and speech signals analyzing. Endowed with these novel capabilities, the mobile robot displays intelligence and cooperativeness in its behavior, and becomes a versatile hardware system for human-robot multimodal communication implementation.

In 2014, researchers proposed a bio-inspired event-driven vision system using pseudo-simultaneous sensing [137]. The control system consists of a Spartan6 FPGA implementing a Convolutional Neural Network and a 3D construction algorithm. Experimental results demonstrate the application of this paradigm to implement Gabor filters and 3D stereo reconstruction systems. This architecture can be applied to real systems which need efficient and high-speed visual perception, like vehicle automatic driving, robotic applications in unstructured environments, or intelligent surveillance in security systems. Experimental results highlight the performance gains of the proposed methodology.

The authors of [147] published a research paper about a hexapod (6-legged spider like) robot using an FPGA-based robotic platform (Spartan 6). The system architecture involves an image sensor controller, a servo-motor controller and an AVR processor. Overall, the implementation of the servo motor controller in an FPGA has offered several advantages in terms of circuit design flexibility and simultaneous command executions when compared to conventional microcontroller-based system.

In 2014, researchers proposed an FPGA-based OV driver for VGA cameras [132]. For their implementation, they used an FPGA board (Cyclone EP2C5T144C8N) and optimized it using VHDL. The results from the conducted experiments suggest that the devised solution outperforms when compared to the coexisting implementations in terms of resources utilization.

Researchers published a research paper implementing a visual sensor unit for mapping and localization using a six-dimensional Micro Aerial Vehicle (MAR - drone) [126]. The sensor design is based on a XILINX FPGA System-on-Chip combining FPGA resources with an ARM A9 Dual-core processor

One year later, a group of researchers proposed a real-time machine vision system for an industrial robot that grasps items from an assembly line [119]. The items are in the same class, meaning that they have the same general shape but different details. The conducted experiments suggest that the machine parts can be located accurately under arbitrary in-plane rotations and can be classified correctly according to the details in their shapes. Guo's system can run with an industrial camera at a resolution of 640×480 and a speed of 50 fps (frames per second) or higher.

In 2015, the authors of an interesting research paper proposed a pragmatic incremental algorithm for achieving gain control and white balancing on an FPGA [112]. The proposed methodology significantly reduces the effects of the variations in lighting on the thresholds. In their paper, they propose a new color space which maximises the hue separation of the different coloured regions to improve the colour discrimination. All of the algorithms are designed to operate on data directly streamed from the camera, enabling a low latency FPGA implementation. The authors conducted real world experiments to evaluate the stability and performance of the proposed prototype.

In 2015, a group of researchers proposed a humanoid robot head with a vision system which has 22 degrees of freedom, including eyebrows sliding mechanism, the eye mechanism, jaw and neck mechanism [108]. The control system of the proposed prototype is an FPGA board. Facial movements are realized through driving traction mechanisms by controlling servos.The vision system of the robot S HFR-II Ihas a human-computer interaction interface with four kinds of work patterns,which includes facial expression recognitions and representations, face recognitions, gender recognitions and expression representations. The authors conducted experiments and the results suggest that the robot can recognize six basic expressions and vividly reproduce them.

Similarly to the aforementioned paper, in the same year, a group of researchers published an implementation of a face recognition system that captures facial images and recognizes a face and detects slight expressional changes of the test image [104]. In this paper, the authors read various facial images and store them. Then, images are compared bit by bit and any mismatch is detected. If the image is matched then a "Match found" message is displayed.

The author of [97] proposed a hardware accelerator to reduce the computation time of the scale-space analysis and the feature detection of the KAZE algorithm on Zynq-SoC family of FPGAs.

An interesting paper is proposed by [91] demonstrates an implementation of image mosaic algorithm on the Xilinx Zynq-7020 FPGA board. The system is able to process more than 60 frames per second (fps) and is capable of maintaining low power consumption. The authors verified the proposed methodology in a synthetic environment simulated using MATLAB.

A real time target tracking and positioning system based on FPGA is proposed by the authors of [89]. The hardware platform for the design is the development board by Xilinx´The system uses monocular camera to measure the orientation of target in real time. It acquires the video stream from a CMOS camera. The system realizes target tracking by extracting the color of the target as the feature, filtering˙detecting feature of the target and then displaying on the screen. The system realizes target positioning according to the centroid of the tracked target, then computing the orientation which can be used for the intelligent robot vision.

A group of researchers proposed a system that integrates different techniques to perform dimensional measurement of thick steel plate bevels using a DE0-Nano FPGA board [88]. The groove modeling is achieved through the use of line detection algorithms that look for straight lines and segments. The system level modeling and synthesis and the integration to the whole digital linear welding system, using only 28% of the FPGA's slices. Results Show the approach is a viable option to automate the process that is currently widely adopted in industry.

An interesting work about a high speed ball target tracking system based on FPGA (ZYNQ) [85] is proposed. With a CMOS sensor, the original frame can be transferred to the FPGA port. After the video streams go through the parallel image processing core, the color and corner features can be extracted. When compared to the predefined target's pattern, the target can be detected. The authors conducted real-world experiments to verify the feasibility of the proposed methodology.

In 2017, a group of researchers published their paper about an FPGA based real-time virtual blind cane system for the visually impaired, detecting obstacles in front of the user in real time [70]. Described three novel architectures of image segmentation, implemented on Virtex UltraScale XCVU190-2FLGC2104EES9847 FPGA. Working as a blind cane, the proposed system helps a blind person find the type of obstacle and the distance to it. The distance from the user to the obstacle is estimated by extracting the laser coordinate points on the obstacle, as well as tracking the system pointing angle. The paper provides a simple method to classify the obstacle's type by analyzing the laser intersection histogram. Real experimental results were also presented in the paper to highlight the validity and accuracy of the proposed system.

The same year, a group of researchers presented three novel architectures of image segmentation, implemented on Virtex UltraScale XCVU190-2FLGC2104EES9847 FPGA [69]. The sequential design which allows control over minute details of arithmetic design, The hybrid architecture provides a full fledged parallel and pipelined implementation of the algorithm and the hybrid architecture overall provides at least two times gain in acceleration.

One year later, a group of researchers propose a real time tracking hardware based on FPGA [60]. They converted a simulated and verified MATLAB code directly into HDL. The key point of their research is that the simulated and hardware outputs were consistent. Three types tracking algorithms (Kalman filter, extended Kalman filter and $\alpha\beta\gamma$ filter) were implemented to validate the proposed methodology.

In 2018, a research paper about a design approach of accelerating the execution of a robust algorithm based on Horn's quaternion solution [59] is proposed. The authors presented a HW architecture as well as its FPGA implementation. In their research paper, they also included results of the experiments they conducted.

Figure 6 depicts the total number of research papers that implemented FPGA Image Processing and the total Image Processing papers in robotics, from 2010 to 2019. It is evident that the popularity of FPGA and Image processing is steadily growing.

4.2. Application Field: Robotic Controller

A group of researchers propose an inverse kinematic algorithm for a parallel milling robot. The proposed robot consists of three robotic arms connected to a moving platform inside which there is a rotating spindle along a trajectory in the cartesian x, y, z space. The system is implemented using Altera Cyclone IV FPGA chip and it is evaluated by experimental results.

A group of researchers propose a robotic platform based on Lego Mindstorms in [162]. In their research, they included hardware, software and mechanical modules. A mathematical model for the motor and the PI controller are included. Experimental results highlight the performance gains of their implementation.

A system architecture controlling and monitoring household appliances through GSM. The system consists of an FPGA Xilinx SP605 implemented in VHDL, the GSM circuit, the PIR sensor and the interface circuit that monitors the appliances

A connecting algorithm for interfacing the FPGA-based controllers to high level robotic software frameworks. The Unity-Link algorithm conjoins PC-processed architectures with nodes that provide demanding real-time control of distributed robotic systems

A group of researchers published their paper about an inverse kinematics FPGA algorithm for a biped robot, based on CORDIC operators of hyperbolic and circular vectoring. The speed of the proposed hardware FPGA application is compared to that of the Nios II CPU. The authors evaluated the proposed methodology by conducting experiments in a synthetic environment.

In 2019, the authors of [25] proposed an FPGA implementation of an adaptive neuro-fuzzy control for a teleoperation system. The controllers were developed in the MATLAB-Simulink environment and were implemented on FPGA using Simulink's fixed point tool and HDL Coder. The experimental results were very promising, as the latencies of these filters on the Field Programmable Gate Array (FPGA) platform were below 300ns with an average latency reduction of 188% (maximum of 570%) over the software versions running on a desktop PC CPU. This open-source event-based filter IP library for FPGA was tested on two different platforms and scenarios using different synthesis and implementation tools for Lattice and Xilinxvendors.
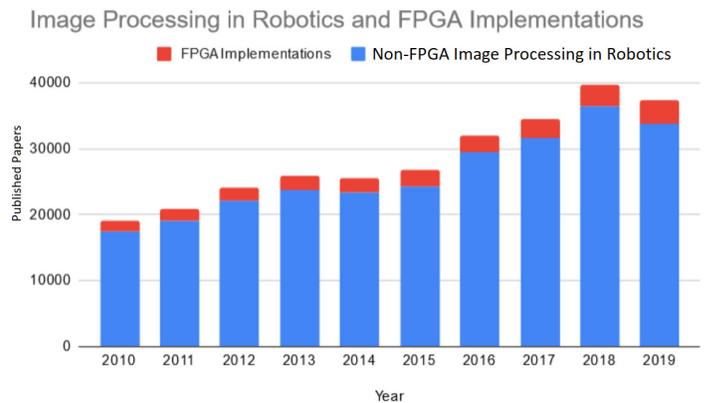
Figure 6. Research papers published about Image Processing using FPGA in Robotics compared to non-FPGA Image Processing papers in Robotics
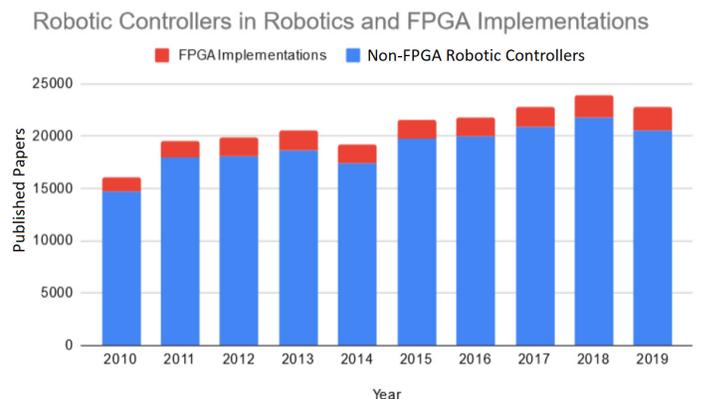
Figure 7. Research papers published about Robotic Controllers using FPGA compared to non-FPGA Robotic Controller papers

Figure 7 depicts the total number of research papers that implemented FPGA Robotic Controllers and the total Robotic Controllers papers in robotics, from 2010 to 2019. It is evident that the popularity of FPGA and Robotic Controllers is increasing.

4.3. Application Field: Neural Network

In 2012, the authors proposed an FPGA-based Neural Network algorithm for lane following of autonomous mobile robots [164]. The neural function consists of two main parts, the fixed points coding and the Sigmoïde function coding. The VHDL simulation is compiled in A Quartus II and implemented in Cyclone EP2C35F484C6 FPGA. The experimental results are promising and suggest high performance gains.

In [163], the authors publish their research paper about a stereo vision landing system architecture for an Unmanned Aerial Vehicle (UAV). The proposed FPGA-based Artificial Neural Network (ANN) algorithm is designed to apply real-time object tracking, 3D position computation via Visual Odometry methodology, Euclidean distance estimation from the landing target and horizontal displacement. The algorithm implementation was done on a Xilinx XC2V100 FPGA chip

In the same year, the authors proposed a Neural Processing Unit implementation [120] which reached a range from 6k to 90k iterations per second for the targeted neural map size. The authors propose a Neural Processing Unit designed as a programmable accelerator implementing recent equations close to self-organizing maps and neural fields. The proposed architecture is validated on FPGAdevices and compared to state of the art solutions. The trade-off proposed by this dedicated but programmable neural processing unit allows us to achieve significant improvements and makes our architecture adapted to many embedded systems.

An alternative architecture that exploits stochastic computation for doing classification with deep belief networks was proposed by the authors of [110] in 2015. The network was trained offline in MATLAB on the MNIST training dataset, and then implemented on an FPGA for classification at variable stochastic computation precision. Experimental results highlight the performance gains of the proposed methodology.
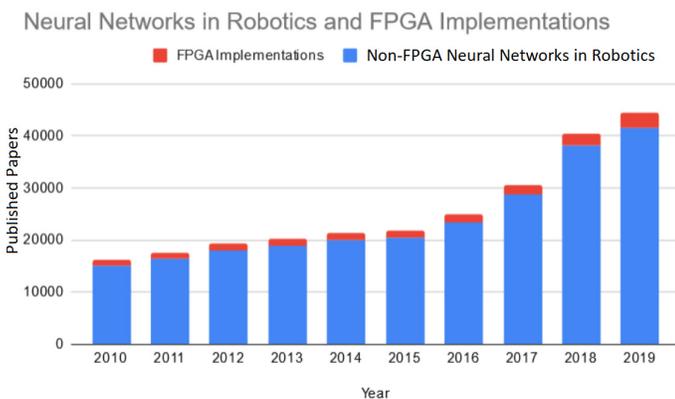

Figure 8. Research papers published about Neural Networks using FPGA in Robotics compared to non-FPGA Neural Network papers in Robotics

Figure 8 depicts the total number of research papers that implemented FPGA Neural Networks and the total Neural Network papers in robotics, from 2010 to 2019. The popularity of FPGA and Networks is increasing steadily. Compared to the early 2010s, the Neural Networks using FPGA and the Neural Networks research papers in general have increased by a factor of 2.92.

4.4. Application Field: Mapping & Navigation

In [146], a color tracking algorithm was proposed for autonomous mobile robots, using a CMOS camera and an ultrasonic sensor. The design implementation was done on a Digilent Nexys2 FPGA. The video is captured and further processed by the FPGA board and displayed on a 3.2 inch screen. They conducted real world experiments to evaluate the stability and performance of their prototype.

A group of researchers proposed a processor embodying a Radio Frequency Identification Detection (RFID) algorithm and a CORDIC algorithm on a Kintex-7 FPGA platform [140]. The proposed system is used for navigation and anti-collision purposes. Experimental results highlight the performance gains of the proposed methodology.

In [135], a Micro Aerial Vehicle (MAV) architecture was proposed consisting of two subsystems: a) An image processing subsystem consisting of two VGA cameras and an Inertial Measurement Unit (IMU) and b) A control system using an Altera SoC FPGA. The interesting aspect of this research is that motion estimation and outlier rejection algorithms were included.

In the same year, an FPGA-based in-door flight localization/navigation system with a pan-tilt camera was proposed [129]. The FPGA (XC6SLX100) was used for image analysis and quad rotor control. The authors validated the proposed methodology by conducting real world experiments.

The authors of [119] proposed an improved ant colony system algorithm for path planning by establishing two new mechanisms for pheromone updating. They included partial pheromone updating and opposite pheromone updating. The authors evaluated the proposed methodology by conducting experiments in a synthetic environment.

[114] presents an effective data cache configuration where the number of active ways and hence its total capacity can be configured at run-time. When using the best data cache configuration, a reduction of 7% the power consumed at a 1% execution time penalty is suggested and it is the same result for the SLAM-EKF algorithm.

The authors of [113] present a novel methodology based entirely on the on-board approach. The obstacle avoidance system that is proposed is suitable for running on low-power embedded devices, leveraging a light-weight low power stereo vision system on FPGA used on Micro Aerial Vehicles (MAVs).

A group of researchers proposed a robot navigation system using a combination of sophisticated image processing algorithms [111]. Their system is used to generate a map and an extended A* algorithm to plan the path. A turtlebot localizes itself in an environment containing several obstacles and a destination. The authors evaluated the proposed methodology by conducting experiments in a synthetic environment.

[109] presents an FPGA-based hardware architecture of EKF Prediction Stage applied to the Mobile Robotics Localization Problem. It runs on an Altera Cyclone IV FPGA with a Nios II processor, adapted to the mobile platform Pioneer 3AT. The experimental results look very promising, as the execution time results point out that the hardware solution has a speed up factor greater than 2 in comparison with the embedded software implementations.

In 2015, a hardware structure for a vision-based autonomous vehicle navigation system was proposed [107]. Results showed that it is possible to implement the structure without consuming a high number of FPGA resources and achieve acceptable execution times. Analyzing the final hardware structure this goal was achieved with a considerably high frame-rate.

In the same year, an investigation was conducted [100]on the amount and sophistication of sensing and processing hardware needed by a mobile robot for performing high-level tasks in real time, such as azimuth calculation and dynamic obstacle tracing. The paper suggests that one ultrasonic sensor along with a coordinate rotation digital computer processor are adequate for azimuth calculation while one more ultrasonic sensor is required for dynamic obstacle tracing.

In [98], a hardware architecture is proposed to accelerate Rapidly-Exploring Random Trees (RRT) path planning algorithm using multiple Kd-trees to conduct the parallel search. The authors evaluated the proposed methodology by conducting experiments in a synthetic environment.

One year later, the authors of [78] propose a technique to evaluate sparse equations on an FPGA by restricting the maximum amount of items in the system. The implementation is done using CλaSH which allows a transformation from mathematical descriptions to a hardware design. The system uses a sparse data notation and puts constraints on the number of loop closures to restrict the amount of computations. Experimental results highlight the performance gains of the proposed methodology.

The first multilevel implementation of the Harris-Stephens corner detector and the Oriented FAST and rotated BRIEF (ORB) feature extractor running on FPGA hardware was published [74] in 2017. The Tarsier device performs excellently as it successfully generates descriptors for real images with sufficiently low latency for integration into a 1080p resolution 60 fps loop.

In the same year, the authors of [62] present their research about the development of a neuromorphic system on a Spartan 6 FPGA generating locomotion patterns (gaits) for three different legged robots (biped, quadruped, and hexapod). The authors validated the proposed methodology by conducting real world experiments. Moreover, the proposed spiking neuron model consumes less hardware resources and power than those based on coupled.

In 2019, an angular position tracking control was implemented using modified D channel PID algorithm [40]. It was developed in such a way that the reading of data, the calculation of the control signals, and the controlling of the motors are all implemented on FPGA. Experimental results highlight the performance gains of the proposed methodology.

Figure 9 depicts the total number of research papers that implemented FPGA Mapping & Navigation and the total Mapping & Navigation papers in robotics, from 2010 to 2019. It is evident that the popularity of FPGA and Mapping is increasing. This could possibly be caused by the general increased popularity of autonomous vehicles (Unmanned Aerial Vehicles, Unmanned Ground Vehicles and Unmanned Underwater Vehicles).
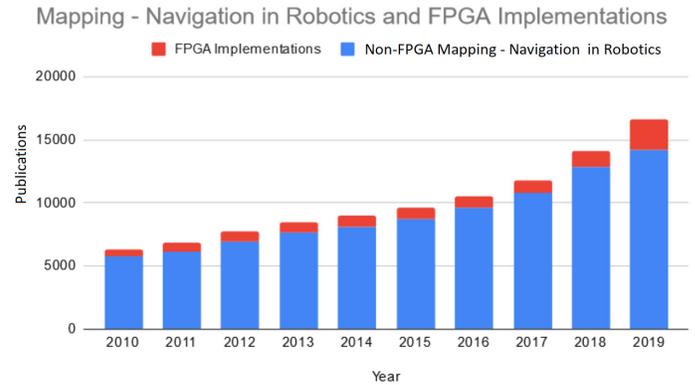

Figure 9. Research papers published about Mapping & Navigation using FPGA in Robotics compared to the non-FPGA Mapping & Navigation papers in Robotics

4.5. Application Field: Fuzzy Logic Control

In 2012 a fuzzy-logic controller [168] for an autonomous navigation robot was proposed. The controller is implemented using an Atmega16 microcontroller and tested using the whole robotic application. The FPGA implementation consists of the Fuzzy logic controller, the UART receiver (it translates data between serial to parallel form), the LCD controller (for controlling the LCD operations) and the ASCII to binary converter and binary to ASCII controllers.

In the same year, an interesting work is proposed about an autonomous parking FPGA-based system for a mobile robot [167]. The system scans the area for available parking space and performs the necessary maneuvers for parking. The whole system consists of three major parts, the car-like robot design, the development of a simulation model using MATLAB and Simulink and the fuzzy logic controller design. The evaluation of the paper was performed using a synthetic realistic environment.

A group of researchers published their research paper about an autonomous mobile robotic platform [165]. Their prototype is designed within the context of a robotic course. The system uses two separate FPGA-based controllers: A Proportional integral derivative (PID) controller and a fuzzy logic-based controller. Apart from the FPGA card, the robot's hardware design consists of two motors, eight sonar sensors, a radio frequency identification reader and a Basic Stamp microcontroller. For the modeling and implementation, they utilized MATLAB and Simulink.

Two fuzzy logic algorithms are developed: one for avoiding obstacles and another for avoiding obstacles / reaching a specific point. For the obstacle avoidance an ultrasonic sensor is used, while software is implemented using LabVIEW modules

In 2013, an interesting work is proposed about a fuzzy logic algorithm that is applied on an autonomous mobile robot for line tracking and obstacle avoidance tasks [145]. The software is

implemented by an Altera Quartus platform while Eclipse software is used to simulate the fuzzy rules. Several practical tests that were conducted by the authors reveal the effectiveness of the proposed fuzzy rules for which MRTQ was able to track the path more precisely and softly at different speeds and avoid the detected obstacle through the path readily.

In 2013, a paper was presented about a fuzzy-logic color extractor used on NI myRIO embedded system package [133]. In total, the fuzzy logic module, the real-time module, and the FPGA LabVIEW module are required for the implementation of the extractor on myRIO embedded device. The authors conducted multiple experiments and evaluated its effectiveness and performance gains.

In 2015, an interesting work is proposed about an omni-directional mobile platform control using the artificial intelligence technique of Fuzzy Logic; the control allows a practical and reliable driving control of 4 omni-directional wheels implemented on FPGA. The experimental results show increased performance and low power consumption.

Figure 10 depicts the total number of research papers that implemented FPGA Fuzzy Logic Control and the total Fuzzy Logic Control papers in robotics, from 2010 to 2019.
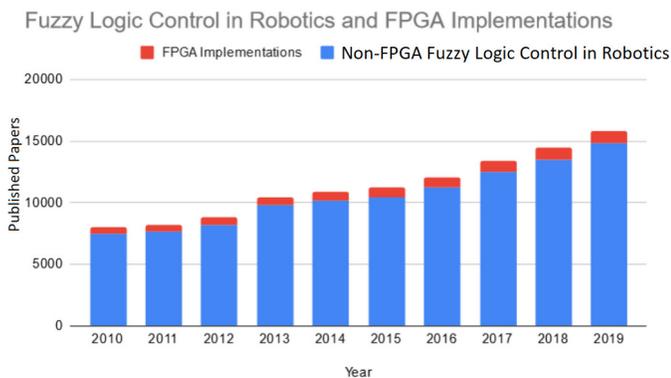

Figure 10. Research papers published about Fuzzy Logic Control using FPGA in Robotics compared to non-FPGA Fuzzy Logic Control papers in Robotics

4.6. Application Field: SWARM

In 2015, an interesting work is proposed about a behavioural control algorithm of a centralized swarm consisting of two robots [101]. In this paper, the authors present the challenges of multi robots and how the leader and follower robots can change their mode of behavioural control by switching between centralization and decentralization methods with a novel approach using implicit communication. According to the authors and their experimental results, the proposed implementation is very energy efficient. Robots are developed with minimal sensors (ultrasonic sensors, digital compass, RF Transceiver (PMOD-RF2)) and are implemented using a Spartan 3e FPGA board

Another interesting implementation was proposed by the authors of [84] in 2014. The heuristic approach based leader follower method is incorporated between robots for behavioural control. The robots were able to navigate in both static and dynamic conditions by using the modified M* based

obstacle avoidance mechanism with CORDIC module. The proposed methodology was validated by the implementation of the algorithm on Spartan 6 FPGAs (XC6SLX45 CSG324C).

In 2015, an interesting work is proposed about a pragmatic incremental algorithm for achieving gain control and white balancing on an FPGA [112]. The proposed methodology significantly reduces the effects of the variations in lighting on the thresholds. In their paper, they propose a new color space which maximises the hue separation of the different coloured regions to improve the colour discrimination. All of the algorithms are designed to operate on data directly streamed from the camera, enabling a low latency FPGA implementation.

Figure 11 depicts the total number of research papers that implemented FPGA SWARM and the total SWARM papers in robotics, from 2010 to 2019. Swarm and FPGA applications in robotics are not as popular as other fields, like mapping, image processing and controllers. However, SWARM and FPGA are still rising in popularity over the years.
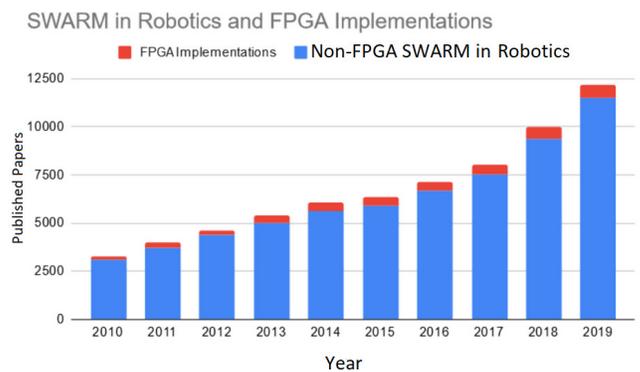

Figure 11. Research papers published about SWARM using FPGA in Robotics compared to non-FPGA SWARM papers in Robotics

5. Discussion

In the 2000s, FPGAs were seen as an inexpensive alternative to the traditional ASICs (Application Specific Integrated Circuit) for heavy computation tasks. Along with the decreased production cost and the technological advancements, this led to the increased popularity of FPGAs. This time period had significant improvements. Most importantly, they were software as well as hardware programmable. In boards released in the late 2000s, for the first time, the microprocessor was not the simple hardware block dropped into the FPGA as was done in older days, but included a full environment with caches, busses, Network-on-Chip and peripherals. Bundled software included operating systems, compilers and middleware: an entire ecosystem, rather than an integrated function block. The increased capabilities of FPGA boards of this era led to the increased usage in robotic applications, such as image processing, robotic controllers, mapping and navigation. In these fields, 10% of the published research papers utilize FPGA boards, presumably for the performance benefits they offer.

In the early and mid 2010s, FPGAs started to gain prominence in another field: neural networks, fuzzy logic control and swarm. Running neural network inference models takes significant

processing power. Graphics processing units (GPUs) are often used to accelerate inference processing, but in some cases, high-performance FPGAs might actually outperform GPUs in analyzing large amounts of data for machine learning.

Just as customers seeking custom integrated circuits 30 years ago were attracted to FPGAs over the complexity of ASICs, many are now attracted to multicore processors, graphic processing units (GPU) and software-programmable Application Specific Standard Products (ASSPs). These alternative solutions provide pre engineered systems with software to simplify mapping problems onto them. They sacrifice some of the flexibility, the performance and the power efficiency of programmable logic for ease-of-use. It is clear that, while there are many FPGA users who need to exploit the limits of FPGA technology, there are many others for whom the technological capability is adequate, but who are daunted by the complexity of using that technology.

The complexity and capability of devices have driven an increase in capability of design tools. Modern FPGA tool-sets include high-level synthesis compilation from C, Cuda and OpenCL to logic or to embedded microprocessors. Vendor-provided libraries of logic and processing functions meet design costs. Working operating systems and hypervisors control FPGA SoC operation. Team modules, including control, are built into FPGA design systems. Some capabilities are built by the vendors themselves, others are part of the growing FPGA ecosystem. Clearly, usability is critical to this next age of FPGAs and their adoption in robotics. If FPGA manufacturers focus on ease of use and solve the steep learning curve, FPGAs will be able to compete with GPUs and become dominant in high performance applications.

6. Conclusions

This survey revealed popular technological, research and industry trends of the past 10 years, regarding robot types used and processed in the research as well as application fields of the FPGAs. As depicted in Figure 4, the vast majority of robots for which applications are being developed are autonomous mobile robots especially when the algorithms developed for these robots are related to localization, mapping and navigation purposes.

In order to further increase the execution speed and efficiency of algorithms, developers utilize FPGA boards, as they offer an innovative solution. The popularity of FPGA boards has been steadily increasing over the last decades, as we can see in Figure 1. FPGAs offer significant advantages over software running on a CPU or GPU, including execution speed and energy efficiency. In a traditional local system, the FPGA board is typically located on-site, along with the other components of the device. However, in modern systems, FPGA boards can be found in the cloud, along with a high performance traditional server. As the server receives data from modern embedded IoT systems, images the server offloads the necessary information to be processed on the FPGA boards. Many boards support PCIe connectivity, with a maximum theoretical bandwidth of 15.75 GB/s in version 3.0. The main disadvantage of FPGA boards, however, is the increased engineering cost and complexity. The development process on FPGA boards includes the design of custom hardware circuits. Traditionally, these hardware circuits are described via Hardware Description Languages (HDL), such as VHDL and Verilog,

whereas software is programmed via one of a plethora of programming languages, such as Java, C and Python. An upcoming trend on FPGA is High Level Synthesis (HLS). HLS is allows the programming of FPGAs using regular programming languages such as OpenCL or C++, allowing for a much higher level of abstraction. However, even when using high level languages, programming FPGAs is still an order of magnitude more difficult than programming instruction based systems. Accelerating the computation process via FPGA boards is only efficient at larger scales, where the performance gain of the system outweighs the increased complexity and engineering cost.

The parallel architectures offered by the FPGAs easily offer the high computational power that continuous real-time localization and navigation tasks demand at a low cost, while mixed FPGA/DSP controllers offer flexibility over the process of complex image data that robotic visual systems include. This particular finding is in accordance with the high number of mapping / navigation application fields shown in Figure 5, for each consecutive year. As shown, the mapping and navigation application field is between half and one third of the total application fields for the 10-years tested period.

As depicted in Figure 4, the field of industrial robotics is the second most usually appearing robot field in the 2010-2019 publication search. FPGAs constitute a flexible solution in the implementation of kinematics algorithms that parallel multi-axis industrial robots often use, in comparison to DSP controllers. The example of the commonly used CORDIC algorithm implemented for rotation tasks in industrial robotics reveals the flexibility of FPGAs. That specific algorithm includes a significant extent of mathematical functions, so the development of a VLSI-based FPGA algorithm is often preferred as the implementation architecture of building a controller for multi-axis motion of robotic limbs. Figure 5 shows the high number of robotic controllers found in the survey, representing the flexibility of FPGAs in industrial robotic control.

Humanoid robots are the third most commonly appeared robot type in the survey. The multi-sensory humanoid robotic applications often demand multi-processor controllers based on mixed DSP/FPGA circuitry. The FPGA connectivity with DSP processors constitutes an effective solution of robotic applications that require multi-processing. The FPGA application fields applied to humanoid robots include regular robotic controllers for motion purposes, image processing applications for the process of visual data for pure image processing or mapping and navigation tasks, or neural network architectures for navigation or else. Swarm robotics also constitutes a wide FPGA application field. Complex behavioural algorithms which demand high computational power and high-levels of real-time communication between swarm members, can be easily implemented through FPGA architectures.

Conflict of Interest

The authors declare no conflict of interest.

Acknowledgment

financed by the European Union and national resources through the B.P. WESTERN GREECE 2014-2020 (project code: DEP5 - 0019433). Table 1 contains other significant works that were reviewed but not fully addressed in section 4.

Table 1: Other significant works that were reviewed

Field	Description	Year	Ref
4.1	A fast frame based on Block RAM reuse	2019	9
4.1	4.1 of a SoC FPGA using RGB-D sensor	2018	21
4.5 / 4.2	A hardware system for on-board support vector machine (SVM) model training for robotic transtibial prostheses locomotion recognition based on SoC-FPGA	2018	22
4.1	FPGA implement for multi-scale pedestrian detection using 4 different size histogram of oriented gradients (HOG) features	2018	23
4.2	An exemplary system for the calculation of the inverse kinematics of a robot arm, which is tailored for the implementation on a FPGA	2018	24
4.5 / 4.2	FPGA-based platform for integrated multi-axis motion control and motor drive	2018	26
4.2	A high-speed, high-accuracy fully pipelined architecture to solve inverse kinematics of a 3-DOF robot leg	2018	27
4.1 / 4.3 / 4.4	SqueezeNet DCNN is accelerated using an SoC FPGA in order for the offered object recognition resource to be employed in a robotic application	2018	28
4.3	A low cost and robust deep-learning based on gaze estimator is proposed to control surgical robots	2018	29
4.5 / 4.2	A novel robot control system based on EtherCAT In order to increase control frequency and expansibility for 7-degree-of-freedom (DoF) light-weight	2018	30
4.2	A control system architecture for knee joint surgery robot based on computer and motion (FPGA) control card	2018	31
4.4 - 4.3	An implementation of mechanical, electronics and software for a man-portable AUV with hybrid architecture	2018	32
4.1	A novel heterogeneous computer-on-module, integrating a Xilinx Zynq SoC and an Adapteva Epiphany multi-core processor	2018	33
4.3	A framework to integrate Deep Learning algorithms on the PYNQ-Z1 at the embedded system level	2018	34
4.2	An open-architecture controller with modularity	2018	35
4.1 - 4.4	A wireless panorama camera development specially for snake robot sensing	2018	36
4.2	A novel distributed SoC-based (PS + PL) impedance controller solution implemented in a robotic hand	2018	37
4.3	Implementation of the You only look once (YOLO) object detector on an FPGA	2018	38
4.1	A design approach of accelerating the execution of a robust algorithm based on Horn's quaternion solution and presents a HW architecture as well as its FPGA implementation	2018	39
4.4	Development of a Triangulation based Navigation Algorithm implemented on FPGA	2017	41

4.2	A mixed FPGA-microprocessor control architecture for a multi-axis industrial robot with vision-based behavior	2017	42
4.2	Robotic application performing auto-detection for high-voltage insulator inspection. The FPGA controller is used for step motor control, communication and signal acquisition	2017	43
4.2	A mixed FPGA-Microcontroller control architecture is included, using a Kawasaki RA10N robot body	2017	44
4.2	A high precision model of minimally invasive surgical robot	2017	45
4.2	A robotic system using a hard processor plus FPGA solution where the FPGA performs all real-time tasks, freeing the processor to run lower-frequency high level control and interface to other devices	2017	46
4.3	An FPGA implementation of Q-learning with Artificial 4.3 (ANN) reducing processing time	2017	47
4.2	A robust method based on a Dual-Core ARM Cortex-A9 System-on-Chip FPGA board that controls the motion of a Cartesian robot	2017	48
4.1 - 4.4	Three novel architectures of image segmentation, implemented on Virtex UltraScale XCVU190-2FLGC2104EES9847 FPGA. The sequential design which allowing control over minute details of arithmetic design, The hybrid architecture provides a full fledged parallel and pipelined implementation of the algorithm and the hybrid architecture overall provides at least 2X gain in acceleration	2017	49
4.1 - 4.4	An FPGA based real-time virtual blind cane system for the visually impaired, detecting obstacles in front of the user in real time. Described three novel architectures of image segmentation, implemented on Virtex UltraScale XCVU190-2FLGC2104EES9847 FPGA	2017	50
4.4	A novel approach of a robot navigation in smart indoor environment developed with the triangulation techniques using Coordinate Rotation Digital Computer (CORDIC) method	2017	51
4.2	Using inverse kinematics and zero moment point algorithmic program (IK-ZMP) to command robotic legs in parallel manner which algorithms were programmed in VHDL	2017	52
4.2	An overall design of the triple inverted pendulum on a cart model which can be used to illustrate various control design problems related to underactuated nonlinear mechanical systems	2017	53
4.4	The first multilevel implementation of the Harris-Stephens corner detector and the Oriented FAST and rotated BRIEF (ORB) feature extractor running on FPGA hardware, for computer vision and robotics applications. The Tarsier device performs excellently-it successfully generates descriptors for real images with sufficiently low latency for integration into a 1080p resolution 60 fps loop	2017	54
4.2	An open and high-performance industrial robot controller which consists of dual MCU + FPGA	2017	55
4.5 - 4.3	A swarm-based algorithm using spiking 4.3 (SNN) in order to simulate biological neuronal assemblies. The FPGA SNN system uses an FPGA Artix-7 chip executing mapping tasks	2017	56
4.3 - 4.2	A cerebellum-like spiking neural network model to be used as the short-range timing function for the talking robot's control system	2017	57
4.4	A technique to evaluate sparse equations on an FPGA by restricting the maximum amount of items in the system. The implementation is done using CλaSH which allows a transformation from mathematical descriptions to a hardware design. The system uses a sparse data notation and puts constraints on the number of loop closures to restrict the amount of computations	2017	58
4.5	A walking robot developed with the FPGA controller for investigating the robot's performance for moving in any direction and avoid obstacles. An FPGA Altera Max II microcontroller has been used, programmed with VHDL and fuzzy logic functions to control the walking robot	2016	61

4.4	An FPGA-based control unit is designed for a patrol robot, using a Spartan 3E board. Depth-first search (DFS) algorithm is used for visiting all the nodes in the patrolling path and this is implemented using black tags (RFID) and IR sensors	2016	63
4.4-4.6	A multi robots path planning like shortest path, behavioral control and obstacle avoidance using a Spartan 6 FPGA with modified M algorithm has resulted with less distance by that it consumes less power	2016	64
4.1	A high speed ball target tracking system based on FPGA (ZYNQ)	2016	65
4.4	A prototype for an autonomous robot using the FPGA DE0-Nano platform to implement. The system designed for having easier maintenance, control and, environmentally friendly	2016	66
4.2	Analyzing and accelerating the inverse Jacobian, pseudo inverse Jacobian, gradient descent, and the Newton method through an FPGA design using high-level synthesis (HLS). Implemented four numerical methods for solving the inverse kinematic problem of a 4 Degrees Of Freedom (DOF) robotic arm	2016	67
4.1	A system that integrates different techniques to perform dimensional measurement of thick steel plate bevels using a DE0-Nano FPGA board. The groove modeling is achieved through the use of line detection algorithms that look for straight lines and segments	2016	68
4.1	An implementation of image mosaic algorithm which is comprised of image registration and image fusion for better effect on the Xilinx Zynq-7020 FPGA board. The system is able to process more than 60 frames per second (fps) and maintaining a low power	2016	71
4.4 - 4.1	A processor-centric FPGA-based architecture for a latency reduction in the vision-based robotic navigation. Also, it can be used to other tasks like detection and tracking of image salient points	2016	72
4.4 - 4.1	A 6 Degree of Freedom (DoF) camera pose estimation algorithm using a monocular camera and non co-planar markers, design for robotics applications, and an implementation on a low cost CPU and the FPGA that can conduct precise 6-DoF estimations at 100Hz	2016	73
4.2	An effective method for calculating the onboard real-time path planning of large-scale space redundant manipulators	2016	75
4.4 - 4.1	A vision-based lane detection algorithm which requires no knowledge of any physical parameters like position and orientation of the camera and is hence very flexible. The algorithm is tested with 14 videos and demonstrate super fast speed, high accuracy, and super energy-efficient	2016	76
4.1	A hardware accelerator to reduces the computation time of the scale-space analysis and the feature detection of the KAZE algorithm on Zynq-SoC family of FPGAs	2016	77
4.3	A design automation tool, named Deep Burning, allowing the application developers to build from scratch learning accelerators that target their specific 4.3 models with custom configurations and optimized performance	2016	79
4.4-4.1	An investigation on the amount and sophistication of sensing and processing hardware needed by a mobile robot for performing high-level tasks in real time such as azimuth calculation and dynamic obstacle tracing which shown that one ultrasonic sensor along with a coordinate rotation digital computer processor are adequate for azimuth calculation while one more ultrasonic sensor is required for dynamic obstacle tracing	2016	80
4.4 - 4.6	Behavioural navigation control algorithm of a centralized swarm consisting of two robots, using FPGA Spartan 3e board	2015	81
4.4 - 4.1	A practical analysis of a SoC architecture that incorporates an FPGA with an embedded ARM processor for Visual SLAM applications	2015	82
4.2	A design and implementation of a novel system for robot-assisted fracture manipulation. The system was evaluated through positioning trials to measure its accuracy and repeatability	2015	83

4.4 - 4.1	A real-time image de-blurring by dynamics-based approach and parallel system architecture	2015	86
4.4 - 4.1	A hardware structure for a vision-based autonomous vehicle navigation system. Results showed that it is possible to implement the structure without consuming a high number of FPGA resources and achieve acceptable execution times. Analyzing the final hardware structure this goal was achieved with a considerably high frame-rate	2015	87
4.3	An alternative architecture that exploits stochastic computation for doing classification with deep belief networks. The network was trained offline in MATLAB on the MNIST training dataset, and then implemented on an FPGA for classification at variable stochastic computation precision	2015	90
4.1 – 4.6	Pragmatic incremental algorithms for achieving gain control and white balancing have been described and implemented on an FPGA	2015	92
4.4 - 4.1	A novel entirely on-board approach, obstacle avoidance system that is suitable for running on low-power embedded devices, leveraging a light-weight low power stereo vision system on FPGA used on Micro Aerial Vehicles (MAVs)	2015	93
4.4	An effective data cache configuration where the number of active ways and hence its total capacity can be configured at run-time. When using the best data cache configuration reveal a reduction of 7% the power consumed at a 1% execution time penalty and it is the same results for the SLAM-EKF algorithm	2015	94
4.2	A finger control for an anthropomorphic piano robot. The robot can play the right note and right rhythm, the punching force of the fingers are controlled just right while the robot is playing	2015	95
4.4	An improved ant colony system algorithm for path planning by establishing two new mechanisms for pheromone updating, including partial pheromone updating and opposite pheromone updating	2015	96
4.1	A real-time machine vision system for industrial robot to grasp from an assembly line a class of machine parts which are similar in the general shape but different in details using FPGA for acceleration	2015	99
4.4 - 4.1	Image-processing navigation algorithm using C language	2014	102
4.2	A dynamic (mathematical) model of a robot finger simulating human motion. An FPGA-based EtherCAT master-slave platform is developed in order to validate the model	2014	105
4.1 – 4.4	A visual sensor unit for mapping/localization used by a six-dimensional Micro Aerial Vehicle (MAR - drone). The sensor design is based on a XILINX FPGA System-on-Chip combining FPGA resources with an ARM A9	2014	106
4.4 - 4.1	A Micro Aerial Vehicle (MAV) architecture consisting of two subsystems: a) An 4.1 subsystem consisting of two VGA cameras and an Inertial Measurement Unit (IMU) and b) A control system using an Altera SoC FPGA. Motion estimation and outlier rejection algorithms are included	2014	115
4.2 - 4.1	A high-speed image visual servoing (VS) system is developed for a SCARA robot. VS is the procedure of the right positioning of the manipulator in order to mitigate the effects of disturbances. The proposed system uses a Cyclone III FPGA controller with a camera and a Position Sensitive Detector (PSD) for position computation	2014	116
4.1 - 4.3	A bio-inspired event-driven vision system not represented by frames, using pseudo-simultaneous sensing. The control system consists of a Spartan6 FPGA implementing a Convolutional Neural Network and a 3D construction algorithm	2014	117

4.2	An FPGA smart sensor interface remotely communicating the main controller with the sensors. An EP3CI6F484 FPGA controller interfaces with SDRAM controller, vibration sensors and a camera. The robot can be used for disaster affected areas and other fields where remote communication is demanded	2014	118
4.2	A Kinematics architecture is applied on an Altera FPGA-based parallel industrial 3-axis milling robot, using the CORDIC algorithm	2013	121
4.2	A Kinematics algorithm is applied on an Altera FPGA-based parallel industrial milling robot, using C language.	2013	122
4.4 - 4.1	An FPGA Based Platform for localization/navigation through a software/hardware solution	2013	123
4.2	A modular robot manipulator consisting of three modules with embedded FPGA controllers, for increased precision. The manipulator uses the Virtual Decomposition Control (VDC) technique where the host computer executes only kinematics calculations, and all the other control calculations are done by local decentralized embedded slave computers	2013	124
4.5	A fuzzy logic algorithm is applied on an autonomous mobile robot for line tracking and obstacle avoidance tasks. The software is implemented by an Altera QuantusII platform while Eclipse software is used to simulate the fuzzy rules	2013	125
4.2	A Spartan-3 FPGA chip is used as a controller for a servomotor control technique implemented for a hexapod robotic platform. The robot consists of 18 servomotors. The controller is programmed with VHDL and the simulation is done with a Xilinx ISE Simulator	2013	127
4.4 - 4.3	A multi-layer neural network perceptron application using a Spartan III FPGA chip designed to drive two servomotors of an autonomous mobile robot. FPGA has been described in VHDL	2013	128
4.2	An ARM + FPGA multi-axis motion controller is implemented on an industrial robotic platform (SCARA robot). The controller is adapted to independent real-time network communication in order to achieve real time servo control	2013	130
4.2	A smoothing robust control algorithm using the Manifold Deformation Design Scheme (MDDS) is implemented for a Delta robot. The controller is a combined Kernel FPGA and DSP. PFGA is used for computation of the feedback signals and synchronously control each joint while DSP performs MDDS calculations	2013	131
Other	A median filter based on Xilinx Spartan-3 FPGA. It is used to filter the output of the measurement module of a line sensor IP core by smoothing the signals, suppress impulse voice and preserve edge	2013	134
4.5	Two fuzzy logic algorithms are developed: one for avoiding obstacles and another for avoiding obstacles / reaching a specific point. For the obstacle avoidance an ultrasonic sensor is used, while software is implemented using LabVIEW modules	2013	136
4.1 - 4.4	A stereovision measurement algorithm based on (Scale Invariant Feature Transform) SIFT key points, implemented on an embedded 4.1 board based on DSP and FPGA. SIFT algorithm looks for key points in the image scale space and extracts invariant locations. The system uses two CCD cameras and an 4.1 board	2013	138
4.2	A connecting algorithm for interfacing the FPGA-based controllers to high level robotic software frameworks. The Unity-Link algorithm conjoins PC-processed architectures with nodes that provide demanding real-time control of distributed robotic systems	2013	139
4.1	A real time 4.1 patrol service robot consisting of a CMOS camera module sending data to the FPGA board	2012	141
4.2 - 4.4	A robotic platform - based on Lego Mindstorms - is presented, including hardware, software and mechanical modules. A mathematical model for the motor and the PI controller are included	2012	142

4.3	A stereo vision landing system architecture for an Unmanned Aerial Vehicle (UAV). The proposed FPGA-based Artificial Neural Network (ANN) algorithm is designed to apply real-time object tracking, 3D position computation via Visual Odometery methodology, Euclidian distance estimation from the landing target and horizontal displacement. The algorithm implementation was done on a Xilinx XC2V100 FPGA chip	2012	143
4.3	An FPGA-based Neural Network algorithm for lane following of autonomous mobile robots. The neural function consists of two main parts, the fixed points coding and the Sigmoïde function coding. The VHDL simulation is compiled in A Quartus II and implemented in Cyclone EP2C35F484C6 FPGA	2012	144
4.5	A fuzzy-logic controller for an autonomous navigation robot. The controller is implemented using an Atmega16 microcontroller and tested using the whole robotic application. The FPGA implementation consists of the 4.5ler, the UART receiver (it translates data between serial to parallel form), the LCD controller (for controlling the LCD operations) and the ASCII to binary converter and binary to ASCII controllers	2012	148
4.3	A hardware application of an FPGA-based controller using an artificial neural approach. The controller is capable of landscape learning, path planning, obstacle avoidance and sensory motor controlling. The controller is implemented as a SoC embedded on the robot, using a Virtex 6 VLX240T FPGA. The basic architecture of the controller consists of three basic layers: the pre-processing layer, the Self Organizing Map layers based on Kohonen maps and the computing layer	2012	150
4.4	A path planning / navigation algorithm for an autonomous mobile robot, using CORDIC architecture for implementing rotation. The robot's external interface includes an FPGA board consisting of Spartan XC2S50 FPGA and three ultrasonic sensors. The coding has been performed using VHDL and synthesized using Xilinx ISE	2012	151
4.1	A multi-resolution real-time dense stereo 4.1 algorithm is implemented. The algorithm's implementation is done on an Altera Stratix IV FPGA board with GigE vision cameras, using six levels of Gaussian Pyramids	2012	152
4.2	A hardware / software robotic application consisting of an operating software system that drives the FPGA controllers. The robotic (5 axe) system is an MK-2 industrial robotic arm and its controllers are operated by a XilinxVirtex II FPGA board and two PowerPC processors	2012	153
4.1	Digital image compression and decompression architecture applying M-JPEG2000 image compression standard, used for an UAV data link system. The two primary components of the system are an FPGA and an ADV212	2012	154
4.2	An FPGA-based speed controller for a DC motor designed for an autonomous mobile robot. The proportional integral (PI) controller used to achieve acceleration realized by the FPGA, while for decelerating control a braked deceleration circuit module is proposed. An additional motor protection circuit to avoid the motor driver IC damage is also included	2012	156
4.3	A 4.2 consisting of a multi-processor System on Chip implemented on a Spartan 3A FPGA, performing bio-inspired walking on a six-ledged robot (hexapod). The whole system consists of one soft processor used for high level decisions in motion decisions and six soft processors running independent control loops for the six legs respectively	2012	157
4.2	A hybrid architecture consisting of parallel FPGA (Spartan 6) and Advance RISC Machine microcontroller for visual trajectory-tracking. The system's framework consists of 3 main layers: The data layer where the FPGA performs input/output extension and hardware acceleration, the real-time layer where the microcontroller performs real-time control and the high performance layer where a computer executes complex tasks like navigation, map generating and artificial intelligence	2012	158
4.4	An FPGA-based algorithm for path planning and obstacle avoidance of autonomous mobile robots. The system consists of ultrasonic sensors and is implemented in a Xilinx FPGA Spartan II	2012	159
	A dexterous six-DOF, five-joint robotic arm for ping-pong is proposed. Each joint contains a motor position sensor, a current sensor and a temperature sensor and an FPGA for communication and control. More		

4.2	specifically, each joint contains four modules, the sensor signal processing, the joint servo control, the motor current control and the communication module	2012	160
4.4	An in-door navigation architecture implemented on FPGA based on impulse-based ultra-wideband (UWB) technology. The system consists of an UWB receiver installed on an autonomous mobile robot that utilizes time-difference on arrival (TDOA) between pairs of anchor nodes for localization purposes. These nodes transmit a coded repeated stream of ultra-short pulses	2011	166
4.5	A locomotion / visual information common function based on fuzzy logic is implemented on an FPGA.	2011	169

References

[1] R. Johannes, M. Porrmann, and U. Rückert. "Survey of FPGA applications in the period 2000–2015" Technical Report, Center of Excellence Cognitive Interncation Technology, Bielefeld University, Germany, 2017.

[2] M. Prado, "An FPGA-based open architecture industrial robot controller" IEEE Access 6, 13407--13417, 2018. .

[3] P. Carlos and M. Porrmann. 2007. The utilization of reconfigurable hardware to implement digital controllers: a review. In 2007 IEEE International Symposium on Industrial Electronics (ISIE'19). 2380--2385.

[4] T. Russell, K. Pocek, and A. DeHon "Reconfigurable computing architectures". In Proceedings of the IEEE. IEEE 103, **3** (2015), 332--354, 2015

[5] V. Anuj, K. Dang Pham, and D. Koch, "A survey on FPGA virtualization", 28th International Conference on Field Programmable Logic and Applications, 1310--1317, 2018..

[6] S. Gandhare and B. Karthikeyan "Survey on FPGA Architecture and Recent Applications" International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN'19), 1--4, 2019.

[7] B. Ari and F. Mondada, "Elements of robotics" Springer Nature, 2018.

[8] W. Mike. 2014. "Implementation of robot systems: an introduction to robotics, automation, and successful systems integration in manufacturing" Butterworth-Heinemann.

[9] H. Martin, "Industrial robotics", Springer handbook of robotics. Springer Nature, 2016

[10] A. Yuki, K. Okada, "Design principles of a human mimetic humanoid: Humanoid platform to study human intelligence and internal body systems". Science Robotics **2**, 13, 2017.

[11] S. Olivier and Thomas Flayols. 2019. An overview of humanoid robots technologies. In Biomechanics of Anthropomorphic Systems. Springer Cham, (2019), 281--310.

[12] S. Sánchez, R. Cortés, "Cartesian control for robot manipulators". In Robot manipulators trends and development, 165--212, 2010

[13] B. Vanderlei, E. Marques, and G. A. Constantinides, "A floating-point extended kalman filter implementation for autonomous mobile robots". Journal of Signal Processing Systems **56**, 1, 41--50, 2019.

[14] G. Mingas, Emmanouil Tsardoulias, and Loukas Petrou. 2012. An FPGA implementation of the SMG-SLAM algorithm. Microprocessors and Microsystems 36, 3, 190--204, 2012

[15] P. Young-seuk, S. Lek. "Artificial Neural Networks: Multilayer Perceptron for Ecological Modeling". Developments in environmental modelling, **28**, 123--140, 2016.

[16] Poliac, M. O., J. M. Zanetti, and D. M. Salerno. 1991. Performance measurements of seismocardiogram interpretation using neural networks. In 1991 Proceedings Computers in Cardiology (IEEE'91). 573--576.

[17] K. Erdal, M. Ahmadieh Khanesar "Fuzzy neural networks for real time control applications: concepts, modeling and algorithms for fast learning", Butterworth-Heinemann, 2016.

[18] P. Yanik, George Ford, and William McDaniel. 2010. An introduction and literature review of fuzzy logic applications for robot motion planning. In Proceedings of ASEE Southeast Section Conference (ASEE'10).

[19] I. Mario and M. Chacón. 2006. Fuzzy logic for image processing: definition and applications of a fuzzy image processing scheme. In Bai Y., Zhuang H., Wang D. (eds) Advanced Fuzzy Logic Technologies in Industrial Applications. 101--113.

[20] S. Giandomenico "Swarm Robotics". MDPI-Multidisciplinary Digital Publishing Institute, 2019

[21] A. Podlubne and D. Göhringer. FPGA-ROS: Methodology to Augment the Robot Operating System with FPGA Designs. In 2019 International Conference on ReConFigurable Computing and FPGAs (ReConFig'19). 1--5, 2019

[22] C. Fu and Y. Yu "FPGA-based Power Efficient Face Detection for Mobile Robots". In 2019 IEEE International Conference on Robotics and Biomimetics (ROBIO'19). 467--473, 2019

[23] T. Sun, Y. Liu, and Y. Wang "Design and implementation of a High-speed Lidar Data Reading System based on FPGA", In 2019 IEEE International Conference on Real-time Computing and Robotics (RCAR'19). 322--327, 2019

[24] K. Hocine, R. Mellah, and H. Talem, "Neuro-fuzzy Control of a Position-Position Teleoperation System Using FPGA". In 2019 24th International Conference on Methods and Models in Automation and Robotics (MMAR'19). 64--69., 2019

[25] L. Alejandro, "Low latency event-based filtering and feature extraction for dynamic vision sensors in real-time FPGA applications", IEEE Access 7, 134926--134942, 2019.

[26] C. Chien, and C. Hsu. HW/SW Co-design and FPGA Acceleration of a Feature-Based Visual Odometry. In 2019 4th International Conference on Robotics and Automation Engineering (ICRAE'19), 148--152, 2019.

[27] R. Martínez Peralta, E. Zamora Gómez, and J. Humberto Sossa Azuela "Efficient FPGA hardware implementation for robot manipulator kinematic modeling using rational trigonometry" IEEE Latin America Transactions, 1524--1536, 2019

[28] P. Christos, T. Nikos, and B. Michael. Design and Implementation of an APSoC-Based Robotic System with Motion Tracking Teleoperation. In 2019 8th International Conference on Modern Circuits and Systems Technologies (MOCAST'19). 1--4, doi: 10.1109/MOCAST.2019.8742037, 2019

[29] C. Zhengdong et al. 2019. Design of HEVC Intra Model Decision Based on Zynq. In 2019 IEEE International Conference on Real-time Computing and Robotics (RCAR'19). 311--315.

[30] A. Kyriakos et al. High Performance Accelerator for CNN Applications. In 2019 29th International Symposium on Power and Timing Modeling, Optimization and Simulation (PATMOS'19). 135--140. doi: 10.1109/PATMOS.2019.8862166.

[31] G. Ce et al. Customisable control policy learning for robotics. In 2019 IEEE 30th International Conference on Application-specific Systems, Architectures and Processors (ASAP'19). 91--98.

[32] B. G. Oliveira and Jorge Lobo. Interactive Demonstration of an Energy Efficient YOLOv3 Implementation in Reconfigurable Logic. In 2019 5th Experiment International Conference (exp. at'19). 235--236.

[33] Z. Chen et al. Eye-to-hand robotic visual tracking based on template matching on FPGAs. IEEE Access **7** (2019), 88870--88880.

[34] H. Chih-Lyang, H. Wu, and W. Hung. Software/Hardware-based hierarchical finite-time sliding-mode control with input saturation for an omnidirectional autonomous mobile robot. IEEE Access **7** (2019), 90254--90267, 2019

[35] D. , Zeyang and L Jing. Real-Time Attitude Estimation of Sigma-Point Kalman Filter via Matrix Operation Accelerator. In 2019 IEEE 13th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoC). IEEE, 342--346, 2019

[36] D. Lee, C. Cheung, and Dan Pritsker. Radar-based Object Classification Using An Artificial Neural Network. In 2019 IEEE National Aerospace and Electronics Conference (NAECON'19). 305--310, 2019

[37] A. Kamaleldin et al. Modular memory system for RISC-V based MPSoCs on Xilinx FPGAs. In 2019 IEEE 13th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoC'19). 68--73, 2019

[38] L. Ning et al. When Deep Learning Meets the Edge: Auto-Masking Deep Neural Networks for Efficient Machine Learning on Edge Devices. In 2019 IEEE 37th International Conference on Computer Design (ICCD'19). 506--514, 2019

[39] R. Tapiador Morales et al. "Spiking Row-by-Row FPGA Multi-Kernel and Multi-Layer Convolution Processor". In 29th International Conference on Field Programmable Logic and Applications (FPL'19). 248--249, 2019

[40] S. Zsolt, and S. Tihamér, Quadcopter control implemented on FPGA. In 2019 IEEE 19th International Symposium on Computational Intelligence and Informatics and 7th IEEE International Conference on Recent Achievements in Mechatronics, Automation, Computer Sciences and Robotics (CINTI-MACRo'19). 000049--000054, 2019

[41] C. Pedro-F et al. A proposal for a SoC FPGA-based image processing in RGB-D sensors for robotics applications. In 2018 IEEE 2nd Colombian Conference on Robotics and Automation (CCRA'18). 1--6, 2018

[42] M. Jingeng et al. Implementing a SoC-FPGA Based Acceleration System for On-Board SVM Training for Robotic Transtibial Prostheses. In 2018 IEEE International Conference on Real-time Computing and Robotics (RCAR'18). 150--155., 2018

[43] W. Ming-Shi and Zhe-Rong Zhang. FPGA implementation of HOG based multi-scale pedestrian detection. In 2018 IEEE International Conference on Applied System Invention (ICASI'18). 1099--1102.

[44] Koepper Alexander and Karsten Berns. Behavior-Based Approach for Calculation of a Robot Arm's Inverse Kinematics on a FPGA. In ISR 2018 50th International Symposium on Robotics (ISR'18). 1--8, 2018

[45] O. Rahnama, D. Frost, O. Miksik and P. Torr.. Real-Time Dense Stereo Matching With ELAS on FPGA-Accelerated Embedded Devices. In IEEE Robotics and Automation Letters **3**, 3 2018

[46] Y. Sun et al. "An SoC-based platform for integrated multi-axis motion control and motor drive" In 2018 International Power Electronics Conference (IPEC-Niigata 2018-ECCE Asia). 560--564

[47] G. Evangelista, Carlos Olaya, and Erick Rodríguez. Fully-pipelined CORDIC-based FPGA Realization for a 3-DOF Hexapod-Leg Inverse Kinematics Calculation. In 2018 WRC Symposium on Advanced Robotics and Automation (WRC SARA'18). 237--242.

[48] P. G. Mousouliotis et al. Expanding a robot's life: Low power object recognition via FPGA-based DCNN deployment. In 2018 7th International Conference on Modern Circuits and Systems Technologies (MOCAST'18). 1--4, 2018

[49] P. Li et al. Efficient and low-cost Deep-Learning based gaze estimator for surgical robot control. In 2018 IEEE International Conference on Real-time Computing and Robotics (RCAR'18). 58--63, 2018

[50] G. Zhang et al. A Control System Design for 7-DoF Light-weight Robot based on EtherCAT Bus. In 2018 IEEE International Conference on Mechatronics and Automation (ICMA'18). 2169--2174, 2018

[51] H. Sheng et al. Design of robot control system for minimally invasive surgery of knee joint. In 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC'18). 2583--2586.

[52] A. Cadena et al. Development of a hybrid autonomous underwater vehicle for benthic monitoring. In 2018 4th International Conference on Control, Automation and Robotics (ICCAR'18). 437--440.

[53] D. Klimeck et al. Resource-efficient Reconfigurable Computer-on-Module for Embedded Vision Applications. In 2018 IEEE 29th International Conference on Application-specific Systems, Architectures and Processors (ASAP'18). 1--4.

[54] L. Stornaiuolo, M. Santambrogio, and Donatella Sciuto. On how to efficiently implement deep learning algorithms on pynq platform. In 2018 IEEE Computer Society Annual Symposium on VLSI (ISVLSI'18). 587--590.

[55] M. Martínez-Prado et al. An FPGA-based open architecture industrial robot controller. IEEE Access 6 (2018), 13407--13417.

[56] L. Chen et al. Teleoperation Snake Robots with Panorama Vision. In 2018 11th International Workshop on Human Friendly Robotics (HFR'18). 54--59.

[57] S. A. Pertuz et al. A Modular and Distributed Impedance Control Architecture on a Chip for a Robotic Hand. In 2018 31st Symposium on Integrated Circuits and Systems Design (SBCCI'18). 1--6.

[58] H. Nakahara, A demonstration of FPGA-based you only look once version2 (YOLOv2). In 2018 28th International Conference on Field Programmable Logic and Applications (FPL'18). 4570--4571.

[59] N. Dimou et al. Parallel Robust Absolute Orientation on FPGA for Vision and Robotics. In 2018 25th IEEE International Conference on Electronics, Circuits and Systems (ICECS'18). 665--668.

[60] G. Haritha et al. Analysis of real-time tracking filters implementation in FPGA. In 2018 IEEE Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER). 158--162.

[61] S. , Yuhei, Acceleration of Publish/Subscribe Messaging in ROS-Compliant FPGA Component. In Proceedings of the 8th International Symposium on Highly Efficient Accelerators and Reconfigurable Technologies (HEART2017). Association for Computing Machinery, New York, NY, USA, Article 13, 1--6.

[62] Y. Z. Jing et al. Design of auto detection system for high-voltage insulator inspection robot. In 2017 International Conference on Robotics and Automation Sciences (ICRAS'17). 16--19, 2017

[63] L. Chen et al.. Development of an industrial robot controller with open architecture. In 2017 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM'17). 754--756, 2017

[64] Z. Jiang et al. Kinematics calculation of minimally invasive surgical robot based on FPGA. In 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO'17). 1726--1730, 2017

[65] Ben P. Jeppesen, et al. An FPGA-based controller for collaborative robotics. In 2017 IEEE 26th International Symposium on Industrial Electronics (ISIE'17). 1067--1072, 2017

[66] P. Reddy Gankidi and Jekan Thangavelautham. FPGA architecture for deep learning and its application to planetary robotics." In 2017 IEEE Aerospace Conference (AeroCon'17). 1--9.

[67] E. S. Fiestas and S. G. Prado, Motion control of a cartesian robot using a dual- core ARM cortex-A9 system-on-chip FPGA. In 2017 Latin American Robotics Symposium (LARS'17) and 2017 Brazilian Symposium on Robotics (SBR'17). 1--6, 2017

[68] R. Nahar, FPGA based parallelized architecture of efficient graph based image segmentation algorithm. In 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO'17). 98--103, 2017

[69] K. Fan et al. Hardware implementation of a virtual blind cane on FPGA. In 2017 IEEE International Conference on Real-time Computing and Robotics (RCAR'17). 344--348, 2017

[70] M. C. Chinnaaiah et al. A versatile autonomous navigation algorithm for smart indoor environments using FPGA based robots. In 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT'17). 1196--1200.

[71] M. Yaqoob Javed et al. Implementation of FPGA based efficient gait controller for a biped robot. In 2017 2nd International Conference on Robotics and Automation Engineering (ICRAE'17). 42--47, 2017

[72] V. Šetka, R. Čečil, and M. Schlegel. Triple inverted pendulum system implementation using a new ARM/FPGA control platform. In 2017 18th International Carpathian Control Conference (ICCC'17). 321--326, 2017

[73] J. Weberruss et al. FPGA acceleration of multilevel ORB feature extraction for computer vision. In 2017 27th International Conference on Field Programmable Logic and Applications (FPL'17). 1--8.

[74] L. Chen, et al. Development of an industrial robot controller with open architecture. In 2017 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM'17). 754--757.

[75] F. Palumbo et al. Feasibility study of real-time spiking neural network simulations on a swarm intelligence based digital architecture. In 2017 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW'17). 247--250.

[76] T. Vo Nhu, and Hideyuki Sawada. Cerebellum-like neural network for short-range timing function of a robotic speaking system. In 2017 3rd

International Conference on Control, Automation and Robotics (ICCAR'17). 184--187.

[77] R. Appel et al. Design-time improvement using a functional approach to specify GraphSLAM with deterministic performance on an FPGA. 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'17). 797--803.

[78] Z. Zhou et al. Mechatronic design of an ankle-foot rehabilitation robot for children with cerebral palsy and preliminary clinical trial. In 2017 IEEE International Conference on Industrial Technology (ICIT'17). 825--830.

[79] S. Chiang et al. 2017. The design of intelligent interactive service robots. In 2017 International Conference on Advanced Robotics and Intelligent Systems (ARIS'17). 100--103.

[80] S. Pullteap. Development of a walking robot by using FPGA controller. In 2016 11th France-Japan & 9th Europe-Asia Congress on Mechatronics (MECATRONICS)/17th International Conference on Research and Education in Mechatronics (REM'16). 054--057.

[81] J. Savage et al. Configurable Mobile Robot Behaviors Implemented on FPGA Based Architectures. In 2016 International Conference on Autonomous Robot Systems and Competitions (ICARSC'16). 317--322.

[82] M. C. Chinnaaiah, et al. A new approach: An FPGA based robot navigation for patrolling in a service environment. In 2016 International Conference on Research Advances in Integrated Navigation Systems (RAINS'16). 1--4.

[83] M. C. Chinnaaiah, et al. A versatile path planning algorithm with behavioural control using FPGA based robots. In 2016 10th International Conference on Intelligent Systems and Control (ISCO'16). 1--6.

[84] C. Lyu et al. High-speed target tracking base on FPGA. In 2016 IEEE International Conference on Real-time Computing and Robotics (RCAR'16). 272--276.

[85] T. Kho et al. Enhance implementation of embedded robot auto-navigation system using fpga for better performance. In 2016 3rd International Conference on Electronic Design (ICED'16). 309--314.

[86] F. Schwiegelshohn, Florian Kästner, and Michael Hübner. FPGA design of numerical methods for the robotic motion control task exploiting high-level synthesis. In 2016 IEEE International Conference on the Science of Electrical Engineering (ICSEE'16). 1--5.

[87] C. Rafael Steffens et al. FPGA Based Sensor Integration and Communication Protocols for Automated Robot Control in Linear Welding. In 2016 XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR'16). 115--120.

[88] C., Chen, et al. Real-time target tracking and positioning on FPGA. In 2016 IEEE International Conference on Real-time Computing and Robotics (RCAR'16). 448--453.

[89] Y. San Woo et al. 2016. Enhance implementation of flying robot auto-navigation system on FPGA for better performance. In 2016 3rd International Conference on Electronic Design (ICED'16). 315--320.

[90] W. Zhou et al. Real-time implementation of panoramic mosaic camera based on FPGA. In 2016 IEEE International Conference on Real-time Computing and Robotics (RCAR'16). 204--209.

[91] P. Čížek, J. Faigl, Low-latency image processing for vision-based navigation systems. In 2016 IEEE International Conference on Robotics and Automation (ICRA'16). 781--786.

[92] R. Konomura and K. Hori. FPGA-based 6-DoF pose estimation with a monocular camera using non co-planer marker and application on micro quadcopter. 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'16). 4250--4257.

[93] K. Boikos. Semi-dense SLAM on an FPGA SoC. In 2016 26th International Conference on Field Programmable Logic and Applications (FPL'16). 1—4, 2016.

[94] J. Yu et al. On-board real-time path planning design for large-scale 7-DOF space manipulator. In 2016 IEEE International Conference on Mechatronics and Automation, Harbin (ICMA'16). 1501--1506.

[95] K. Huang et al. 2016. A scalable lane detection algorithm on COTSs with OpenCL. In 2016 Design, Automation & Test in Europe Conference & Exhibition (DATE'16). 229--232.

[96] L. Kalms, A. Elhossini, and B. Juurlink. FPGA based hardware accelerator for KAZE feature extraction algorithm. In 2016 International Conference on Field-Programmable Technology (FPT'16). 281--284.

[97] S. Xiao, A. Postula, and N. Bergmann. Optimal random sampling based path planning on FPGAs. In 2016 26th International Conference on Field Programmable Logic and Applications (FPL'16). 1--2.

[98] Y. Wang et al. DeepBurning: automatic generation of FPGA-based learning accelerators for the neural network family. In 2016 53nd ACM/EDAC/IEEE Design Automation Conference (DAC'16).1--6.

[99] P. Vyas et al. Cordic-based azimuth calculation and obstacle tracing via optimal sensor placement on a mobile robot. In IEEE/ASME Transactions on Mechatronics 21, 5 (2015), 2317--2329.

[100] M. C. Chinnaiah, A novel approach in navigation of FPGA robots in robust indoor environment. In 2015 International Conference on Advanced Robotics and Intelligent Systems (ARIS'15). 1--6.

[101] V. Hugo Schulz, A SoC with FPGA Landmark Acquisition System for Binocular Visual SLAM. In 2015 12th Latin American Robotics Symposium and 2015 3rd Brazilian Symposium on Robotics (LARS-SBR'15). 336--341.

[102] G. Dagnino et al. Design and real-time control of a robotic system for fracture manipulation. In 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC'15). 4865--4868.

[103] R. Saumyarup et al. Face detection system using FPGA. In 2015 Online International Conference on Green Engineering and Technologies (IC-GET'15). 12--15.

[104] G. Quintal et al. Real-time FPGA decentralized inverse optimal neural control for a shrimp robot. In 2015 10th System of Systems Engineering Conference (SoSE'15). 250--255.

[105] M. D. Kim and Jun Ueda. 2015. Real-time image de-blurring and image processing for a robotic vision system. In 2015 IEEE International Conference on Robotics and Automation (ICRA'15). 1899--1904.

[106] M. A. Dias and Fernando S. Osório. 2015. Reconfigurable hardware architecture for vision-based driving systems. In 2015 12th Latin American Robotics Symposium and 2015 3rd Brazilian Symposium on Robotics (LARS-SBR'15). 97--102.

[107] X. Ke, Vision development of humanoid head robot SHFR-III. In 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO'15). 1590--1595.

[108] L. Contreras et al.. Hardware architecture of the EKF prediction stage applied to mobile robot localization. In 2015 IEEE 6th Latin American Symposium on Circuits & Systems (LASCAS'15). 1--4.

[109] K. Sanni et al. Fpga implementation of a deep belief network architecture for character recognition using stochastic computation. In 2015 49th Annual Conference on Information Sciences and Systems (CISS'15). 1--5.

[110] J. Rettkowski, Robot navigation based on an efficient combination of an extended A* algorithm, bird's eye view and image stitching. In 2015 Conference on Design and Architectures for Signal and Image Processing (DASIP'15). 1--8.

[111] D. Bailey, Miguel Contreras, and Gourab Sen Gupta. Towards automatic colour segmentation for robot soccer. In 2015 6th International Conference on Automation, Robotics and Applications (ICARA'15). 478--483.

[112] H. Oleynikova, D. Honegger, and M. Pollefeys. Reactive avoidance using embedded stereo vision for MAV flight. In 2015 IEEE International Conference on Robotics and Automation (ICRA'15). 50--56.

[113] B. A. Silva et al. Run-time cache configuration for the LEON-3 embedded processor. In Proceedings of the 28th Symposium on Integrated Circuits and Systems Design (SBCCI'15). 1--6.

[114] Y. Li and C. Huang. Force control for the fingers of the piano playing robot—A gain switched approach. In 2015 IEEE 11th International Conference on Power Electronics and Drive Systems (PEDS'15) 265--270.

[115] C. Hsu et al. FPGA-based path planning using improved Ant Colony Optimization Algorithm. In 2015 IEEE 5th International Conference on Consumer Electronics-Berlin (ICCE-Berlin'15). 443--444.

[116] M. Pena et al . Fuzzy logic for omni directional mobile platform control displacement using FPGA and bluetooth. In IEEE Latin America Transactions 13, 6, 2015, 1907--1914.

[117] L. Zouari, M. Ben Ayed, and M. Abid. A hardware in the loop simulation for electrically driven robot manipulator. In 2015 IEEE 28th Canadian Conference on Electrical and Computer Engineering (CCECE'15). 1495--1501.

[118] H. Guo et al. Real-time detection and classification of machine parts with embedded system for industrial robot grasping. In 2015 IEEE International Conference on Mechatronics and Automation (ICMA'15). 1691--1696.

[119] L. Fiack, L. Rodriguez, and B. Miramond. 2015. Hardware design of a neural processing unit for bio-inspired computing. In 2015 IEEE 13th International New Circuits and Systems Conference (NEWCAS'15). 1--4.

[120] M. C. Chinnaaiah et al. 2014. A novel approach in implementation of rendezvous behavioral control between FPGA robots. In 2014 International Conference on Advanced Robotics and Intelligent Systems (ARIS'14). 97--102

[121] J. Heshmatpanah, S. Boroumand, and Mehdi Tale Masouleh. 2014. FPGA design and implementation for real time vision applications on NTACO mobile robot. In 2014 Second RSI/ISM International Conference on Robotics and Mechatronics (ICRoM'14). 172--177.

[122] A. M. Ashir, Atef A. Ata, and M. Shukri Salman. 2014. FPGA-based image processing system for Quality Control and Palletization applications. In 2014 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC'14). 285--290.

[123] S. Agrawal and Ravi N. Prakash.. Implementation of WSN which can simultaneously monitor Temperature conditions and control robot for positional accuracy. In 2014 International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE'14). 1--6.

[124] M. Hou et al. Analysis of the multi-finger dynamics for robot hand system based on EtherCAT. In 2014 10th International Conference on Natural Computation (ICNC'14). 1061--1065.

[125] J. Nikolic et al. A synchronized visual-inertial sensor system with FPGA pre-processing for accurate real-time SLAM. In 2014 IEEE international conference on robotics and automation (ICRA'14). 431--437.

[126] S. Seok et al. A highly parallelized control system platform architecture using multicore CPU and FPGA for multi-DoF robots. In 2014 IEEE International Conference on Robotics and Automation (ICRA'14). 5414--5419.

[127] F. Mota Muñoz, S. Martínez de la Piedra, and Karla C. Gómez. Development of a multi-DOF robotic controller for academic purposes. In 2014 International Conference on Mechatronics, Electronics and Automotive Engineering (ICMEAE'14). 104--109.

[128] R. Ladig and K. Shimonomura.. Fpga-based fast response image analysis for autonomous or semi-autonomous indoor flight. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW'14). 668--673.

[129] C. Guo et al. Reconfigurable fault-tolerant joint control system for space robot. In Proceeding of the 11th World Congress on Intelligent Control and Automation (WCICA'14). 1899--1904.

[130] T. Zhang et al. Design and control of a multisensory five-finger prosthetic hand. In Proceeding of the 11th World Congress on Intelligent Control and Automation (WCICA'14). 3327--3332.

[131] E. Yusifli et al. An FPGA Based Resources Efficient Solution for the OmniVision Digital VGA Cameras Family. 2014 IEEE Intl Conf on High Performance Computing and Communications, 2014 IEEE 6th Intl Symp on Cyberspace Safety and Security, 2014 IEEE 11th Intl Conf on Embedded Software and Syst (HPCC'14, CSS'14, ICESS'14). 558--561.

[132] D. Oswald et al. Implementation of fuzzy color extractor on ni myrio embedded device. In 2014 International Conference on Multisensor Fusion and Information Integration for Intelligent Systems (MFI'14). 1--6.

[133] C. Andres Lara-Nino, Cesar Torres-Huitzil, and Jose Hugo Barron-Zambrano. Versatile educational and research robotic platform based on reconfigurable hardware. In 2014 International Conference on ReConFigurable Computing and FPGAs (ReConFig14). 1--6.

[134] G. Zhou et al. 2014. On-board inertial-assisted visual odometer on an embedded system. In 2014 IEEE International Conference on Robotics and Automation (ICRA'14). 2602--2608.

[135] M. H. Liyanage and N. Krouglicof . A single time scale visual servoing system for a high speed SCARA type robotic arm. In 2014 IEEE International Conference on Robotics and Automation (ICRA'14). 4153--4160.

[136] L. Alejandro, Teresa B. Linares-Barranco. Event-driven sensing and processing for high-speed robotic vision. In 2014 IEEE Biomedical Circuits and Systems Conference (BioCAS'14) Proceedings. 516--519.

[137] B. Y. Kumar and S. Ilaiyaraja. 2014. Smart interface for sensors in robot. In 2014 Sixth International Conference on Advanced Computing (ICoAC'14). 145--149.

[138] H. B. et al. Embedded FPGA controller for robot arm in material handling using reconfigurable nicompact RIO. In Second International Conference on Current Trends In Engineering and Technology (ICCTET'14). 125--131.

[139] S. Bhashis Roy, J. Bag, and S. Kumar Sarkar. Design and VLSI implementation of a robot navigation processor deploying CORDIC based anti-collision algorithm with RFID technology. In 2014 Annual IEEE India Conference (INDICON'14). 1--6.

[140] M. Petko et al. Acceleration of parallel robot trajectory generation in FPGA. 2013 IEEE 8th Conference on Industrial Electronics and Applications (ICIEA'13). 1123--1128.

[141] M. Petko et al. Acceleration of parallel robot trajectory generation in FPGA. In 2013 IEEE 8th Conference on Industrial Electronics and Applications (ICIEA'13). 1123--1128.

[142] A. Ghorbel et al. An FPGA based platform for real time robot localization. In 2013 International Conference on Individual and Collective Behaviors in Robotics (ICBR'13). 56--61.

[143] W. Zhu et al. Precision control of modular robot manipulators: The VDC approach with embedded FPGA. In IEEE Transactions on Robotics 29, 5 (2013), 1162--1179.

[144] S. Boroumand et al. Path tracking and obstacle avoidance of a FPGA-based mobile robot (MRTQ) via fuzzy algorithm. In 2013 13th Iranian Conference on Fuzzy Systems (IFSC'13). 1--5.

[145] S. Bold, Min A. Jeong, S. Min Jeon, and S. Ro Lee. FPGA based real time embedded color tracking mobile robot. In 2013 International Conference on IT Convergence and Security (ICITCS'13). 1--4.

[146] N. Alisa Ali et al. PWM controller design of a hexapod robot using FPGA. In 2013 IEEE International Conference on Control System, Computing and Engineering (ICCSCE'13). 310--314.

[147] B. Velasquez et al. Navigation control in FPGA for a differential-drive mobile robot. In 2013 16th International Conference on Advanced Robotics (ICAR'13). 1--6.

[148] C. Szász and G. Husi. Novel multimodal communication skills implementation on the NI-9631 robot. In IECON 2013-39th Annual Conference of the IEEE Industrial Electronics Society (IECON'13). 7837--7842.

[149] R. Li, Shiyao Lin, and Jian Chen. Embedded Motion Controller Design Based on RTEX Network. In 2013 5th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC'13). 326--329.

[150] H. Lin "The implementation of smoothing robust control for a delta robot" In 2013 Second International Conference on Robot, Vision and Signal Processing (RVSP'13). 208--213.

[151] Yen-Fang Li and L. Chuang. Controller design for music playing robot—Applied to the anthropomorphic piano robot. In 2013 IEEE 10th International Conference on Power Electronics and Drive Systems (PEDS'13). 968--973.

[152] C. C. Wong and C. C. Liu. FPGA realisation of inverse kinematics for biped robot based on CORDIC. Electronics letters 49, 5 (2013), 332--334.

[153] L. Mutauranwa and Magripa Nleya. 2013. An efficient median filter in a robot sensor soft ip-core. In 2013 Africon Pointe-Aux-Piments (Africon'13). 1--5.

[154] M. Fatih Aydogdu, M. Fatih Demirci, and C. Kasnakoglu. Pipelining Harris corner detection with a tiny FPGA for a mobile robot. In 2013 IEEE International Conference on Robotics and Biomimetics (ROBIO'13). 2177--2184.

[155] M. Mysorewala, et al. Design and implementation of fuzzy-logic based obstacle-avoidance and target-reaching algorithms on NI's embedded-FPGA robotic platform. In 2013 IEEE Symposium on Computational Intelligence in Control and Automation (CICA'13). 168--173.

[156] S. Cruz et al. FPGA implementation of a sequential extended Kalman filter algorithm applied to mobile robotics localization problems. In 2013 IEEE 4th Latin American Symposium on Circuits and Systems (LASCAS'13). 1--4.

[157] H. Song et al. A fast stereovision measurement algorithm based on SIFT keypoints for mobile robots. In 2013 IEEE International Conference on Mechatronics and Automation (ICMA'13). 1743--1748.

[158] A. Blaabjerg Lange, Ulrik Pagh Schultz, and Anders Stengaard Soerensen. Unity-link: A software-gateware interface for rapid prototyping of experimental robot controllers on fpgas. In 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'13). 3899--3906.

[159] B. Thanh Trung and N. Van Cuong. Monitoring and controlling devices system by GPRS on FPGA platform. In 2013 International Conference on Advanced Technologies for Communications (ATC'13). 713--717.

[160] K. Asami, H. Hagiwara, and M. Komori. Visual navigation system based on evolutionary computation on FPGA for patrol service robot. In The 1st IEEE Global Conference on Consumer Electronics 2012 (GCCE'12). 295--298.

[161] M. D.A. Siqueira, J. Conforto, and Marcos B.R. Vallim. An FPGA-Based Robotics Platform for Educational and Research Applications. In 2012 Brazilian Robotics Symposium and Latin American Robotics Symposium (SBR'12, LARS'12). 313--318.

[162] A. Din "Embedded low power controller for autonomous landing of UAV using artificial neural network" In 2012 10th International Conference on Frontiers of Information Technology (FIT'12). 196--203.

[163] A. Benjemmaa. Implementations approches of neural networks lane following system. In 2012 16th IEEE Mediterranean Electrotechnical Conference (MELECON'12). 515--518.

[164] R. Alba-Flores. Project-based learning approach in a senior level course in autonomous mobile robots. In 2012 Proceedings of IEEE Southeastcon (Southeastcon'12). 1--8.

[165] K. Gac. FPGA based hardware accelerator for calculations of the parallel robot inverse kinematics. In Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies & Factory Automation (ETFA'12). 1--4.

[166] N. Scicluna et al. FPGA-based autonomous parking of a car-like robot using Fuzzy Logic Control. In 2012 19th IEEE International Conference on Electronics, Circuits, and Systems (ICECS'12). 229--232.

[167] T. Muthu et al. 2012. Fuzzy logic controller for autonomous navigation. In 2012 International Conference on Communication and Signal Processing (ICCSP'12). 81--92.

[168] H. Hagiwara, K. Asami, and Mochimitsu Komori. Real-time image processing system by using FPGA for service robots. In The 1st IEEE Global Conference on Consumer Electronics 2012 (GCCE'12). 720--723.