To the question of realization of machine vision technology based on FPGA-architectures

K A Timakov
Department of Corporate Information Systems, MIREA – Russian Technological University, 78 Vernadsky ave., Moscow, 119454, Russia
E-mail: Tim1997.10@yandex.ru

Abstract. In the last few years, machine learning and machine vision technologies have started to gain more and more popularity. This industry occupies one of the leading positions in the field of information technology. The paper is devoted to the development of a machine vision algorithm based on new generations of FPGAs for recognizing handwritten Cyrillic characters in images and video streams, in particular. The article raises the issues of using FPGA as an image segmentation accelerator and organizing work with the video stream, choosing the most suitable FPGA platform, creating training samples of handwritten characters, and working with the convolutional neural network AlexNet.

1. Introduction
Computer vision is an area of synthetic intelligence related to the analysis of images and videos. It includes a set of methods that give the computer the ability to "see" and extract information from what it sees.

The systems consist of a photo or video camera and specialized software that identifies and classifies objects. They are able to analyze images (photos, pictures, videos, barcodes), as well as faces and emotions.

To teach a computer to "see", machine-learning technologies are used. A lot of data is collected that allows you to identify features and combinations of features for further identification of similar objects.

Basically, high-quality object recognition by a neural network may require high computing power to process a large amount of data. In order to get around this problem, you can use FPGA accelerators. This solution allows you to increase the processing power of the workstation several times.

2. Analysis of some innovative FPGA technologies
Currently, there are many manufacturers on the market of programmable logic chips, the main ones are: Xilinx, Intel, Atmel, etc. Over the past few years, the market has undergone significant changes, new developments and directions are appearing.
The Intel Agilex family combines several innovative Intel technologies, including the second-generation HyperFlex FPGA matrix, built on Intel's 10nm process, and the diverse 3D technology in the SiP package, based on Intel's proven Multi-Dieb Connecting Bridge technology. This combination of advanced technologies allows Intel to combine the analog unit, memory, custom computing, custom I/O modules, and Intel eASIC devices into a single package along with the FPGA matrix.

The Agilex FPGA is based on the idea of chiplets – a constructor assembled from basic and specialized Intel blocks. The FPGA array is produced according to the most modern 10-nm process technology. Compared to the previous generation of FPGAs, Intel provides up to 40% increase in performance, consumption savings and performance – up to 40 teraflops.

Table 1. Analysis and comparison of Intel Agilex FPGA series.

| Model        | F-Series | I-Series | M-Series |
|--------------|----------|----------|----------|
| Target       | for a wide range of applications | for tasks with high throughput | for high-performance computing |
| Data transfer rate | 58 G | 112G | 112G |
| Model        | F-Series | I-Series | M-Series |
| Generation PCI-e | 4 | 5 | 5 |
| Memory       | DDR4 | DDR4 | DDR4, DDR5 |
| Processor    | Quad-Core ARM Cortex – A53 (SoC option) | Quad-Core ARM Cortex – A53 | Quad-Core ARM Cortex – A53 |
| Scalable Processor option | – | + | + |
| High-bandwidth memory | – | – | + |

One of the most important advantages of using Agilex chips, used in all three series, is the processing of requests to the deep learning network. This process consists of sending sensor data to a trained deep learning network, after which the result should be obtained as quickly as possible. FPGAs can contain configurable DSP/DSP for FP32, BFloat16, FP16, or INT8 processing. Less precise data formats can also be processed, although in practice they are not as widespread [1, 2].

Versal's Xilinx FPGA, the industry's first Adaptive Computing Acceleration (ACAP) platform, combines adaptable processing and acceleration processors with programmable logic and configurable connectivity to create specialized, heterogeneous hardware solutions for a wide range of applications in the data center, automotive and defense industries, 5G networks, and information security.

The Xilinx Versal architecture built on a 7 nm process is the industry's first heterogeneous platform for accelerating computing, which is known as the ACAP-Adaptive Compute Acceleration Platform - an adaptive platform for accelerating computing of any application, combining several different mechanisms at the same time: scalar computing modules, vector processing modules, NoC – Network-on-Chip and other innovative solutions. Together, this allows you to achieve a 90-fold advantage over 16/12-nm solutions on the CPU and GPU.
Table 2. Technical characteristics of the Versal series FPGA boards.

| FPGA resources and capabilities | AI Core Series | Prime Series | Premium Series |
|---------------------------------|---------------|--------------|---------------|
| Programmable network on a chip  | +             | +            | +             |
| Integer operations (flop/s)      | 57-176        | 8-57         | 36-206        |
| System logical cells (ths.d.)    | 540-1.968     | 352-2.233    | 1.575-7.352   |
| Hierarchical Memory (Mb)         | 68-191        | 40-324       | 198-994       |
| Digital signal processing modules DSP58 | 928-1.968     | 472-3.984    | 1.904-14.352  |
| VLIW SIMD AI core accelerators  | 128-400       | -            | -             |
| Processor subsystem              | +             | +            | +             |
| High-speed transceivers          | 8-44          | 12-48        | 72-168        |
| Total throughput (TB/s)          | 2.9           | 5.2          | 18.1          |
| Programmable outputs             | 478-770       | 316-748      | 500-780       |
| DDR Memory Controllers           | 2-4           | 1-4          | 3-4           |

During the analysis of the market of the presented solutions at the moment, it was decided to consider two representatives of Intel and Xilinx for this study. For comparison, two specific models are selected, these are Xilinx Versal Premium and Intel Agilex I-Series. These solutions, due to their capabilities and characteristics, are most suitable for this work. Table 3 provides an analysis and comparison of FPGA families [3, 4].

Table 3. Comparison of FPGA families.

| Characteristics                      | Xilinx Versal Premium | Intel Agilex I-Series |
|--------------------------------------|-----------------------|-----------------------|
| Number of devices/enclosures         | 7/6                   | 2/3                   |
| Technical process, nm.               | 7                     | 10                    |
| Total bandwidth of the transceivers, Tbit/s | 18.0                   | 3.5                   |
| Number of logical elements, mln.     | 1.5-7.4               | 2.2-2.7               |
| 112 Gbps Transceivers (max)          | 70                    | 8                     |
| Ports Ethernet MAC 10/25/50/100/200/400G | -/-/16/50/21/7       | 2/2/2/2/2/2          |
| 600G Interlaken / 600GE MAC)         | 3/7                   | No/2 (400 GE)         |
| PCIe Gen5 (max.)                     | 8 (x4) 2 (x8)         | 3 (x16)               |
| Network on the chip (NoC)            | +                     | -                     |
| Processor system ARM                 | Cortex- A72+Cortex-R5F| 288 (Block RAM, LUTRAM) |
| Available internal hierarchical memory, Mbit | 994 (URAM, Block RAM, LUTRAM) | 288 (Block RAM, LUTRAM) |
| DSP-enabled FP32 TFlops / INT8 Tops performance | 23.0 / 99.0 | 11.8 / 23.6 |

Based on the data from the table, we can conclude that the prevailing number of characteristics, Versal Premium is better than the solution from Intel. Among the superior parameters, the most significant ones can be identified - throughput and the number of logical elements. These parameters are key to evaluating performance.
In this case, it is not necessary to focus on the difference in the technical process, since it does not have a critical value when choosing, because due to the many technological processes of both companies, the density of the elements does not have a significant difference.

Summing up this question, we can conclude that for the implementation of this work, the most suitable solution is the product of the company Xilinx. Xilinx Versal Premium is designed for working with a large data stream, since this board will be used for processing the video stream, this parameter can be key in choosing.

3. Designing a machine vision system based on the Alex Net neural network

After a detailed study and analysis of the existing architectures of convolutional neural networks, the decision was made to use the AlexNet neural network as the most suitable for the task. The convolutional neural network Alex Net has a structure consisting of eight deep layers. This network works with color images with a size of 227x227 pixels. To determine the color, 3 bits are allocated [5].

The versatility of the final product increases significantly when using a pre-trained neural network. A pre-trained network can be used as a reference point to implement a solution to a new problem.

To train a neural network, Matlab software is used, using transfer learning. The advantage of transfer learning is that you can take a pre-trained network and use it as a starting point to learn a new task. Fine-tuning a network with transmission training is usually much faster and easier than training a network from scratch with arbitrarily initialized weights.

The adequacy and quality of the final results during the training of a neural network can be ensured by the representativeness of the initial data, i.e. that the initial data sample correlates with all the characteristic features and properties inherent in the object under study, which would allow us to correctly solve the problem of object classification [6].

To train the neural network, a specially created set of 660 images of various Cyrillic characters was used, 70% of which is used for training, and 30% - for testing and validation. Each image has the same size, namely 227x227 pixels. Figure 1 shows an example of a test sample for the letter «Z».

![Figure 1. Example of a test sample for a letter «Z».

The training data was divided into two categories: a training sample consisting of two-thirds of all images, and a sample for validation of network quality control during training, representing one-third of the images.

Figures 2 and 3 show the graphs obtained during the training of the neural network using Matlab. These graphs show that the learning process was divided into six epochs of 46 iterations each. An iteration is a counter that increases its score when passing one training set. An epoch is a package of training sets, when training a complete package of sets, training begins on a new package (epoch), so it happens six times. The neural network is trained during each iteration, then it conducts testing, and the results of testing are presented on the graph as a black dot. At the end of the full testing cycle, the system outputs the network training result as a percentage.
The convolutional neural network model used in this work showed the recognition accuracy on the test set 93.9%.

After the training of the neural network is completed, the system randomly from the validation set presents a set of recognized characters with a percentage of confidence in the correct answer. Figure 4 shows the result of the network operation.
When recognizing characters in a video stream, there are many factors that reduce the quality of recognition. One of these factors is the poor illumination of the object of interest, the low resolution of the camera, and the unworthy sharpness of the image. When trying to recreate the optimal conditions, the character recognition process in the video stream showed a high percentage of recognition based on the results of testing. Figure 5 shows the result of character recognition in the video stream.

To use the resulting neural network to recognize the input characters and words from the video stream, it is necessary to perform preliminary segmentation. This process can be extremely difficult for a single computer, which ultimately will lead to a significant reduction in the operating time of the neural network as a whole.

One of the main problems when segmenting images and video streams is poor performance. It means that the implementation of many image processing algorithms relies on the repeated use of simple arithmetic operations (summation, multiplication and accumulation), which leads to an increase in computational costs.

To implement digital image processing functions based on modern FPGA devices, it is possible to use a large number of logic blocks, registers and specialized components, such as memory cells and multiplexers. Modern FPGA architecture provides for the presence of an integrated microprocessor, which allows you to optimize the solution of the problem of image processing on the FPGA, depending on the size of the images and the operation performed. Optimization of the solution of the problem of image processing on the FPGA is achieved due to the ability to reconfigure the coprocessor built into the FPGA architecture [7, 8].
Figure 6 shows a simplified scheme for capturing and processing video on an FPGA. Video capture is performed from a camera connected to a computer via a video capture card to maximize data transfer latency. The video stream from the camera is sent to the PC, divided into three channels in the YCbCr color space. All three channels are sent to the module responsible for changing the frame size to the reference size for processing by the segmentation module and the operation of the neural network in the future. Next, the video stream is storyboarded, which is sent to the FPGA frame-by-frame, the frames are accompanied by a synchronization signal for monitoring frames and recording service information. The processing of this stream is performed by the central block "Image Segmentation". The processed frames are sent back to the MATLAB development environment, where the frames are converted back to the video stream, but in a black-and-white representation with the boundaries of the likely objects of interest clearly highlighted. After that, it is sent to the "Color Model Conversion" block, where the video stream from the YCbCr color model is converted to an RGB color model suitable for recognition by a convolutional neural network. At the end, the processed video stream is sent to the input of the neural network.

The implementation of the video stream processing module provides acceleration of the algorithm as a whole. The intelligent system accelerates the solution of the problem of digital image processing. A new generation of programmable logic inside an adaptable system accelerates parallelizable algorithms.

Table 4 shows a comparison of the performance gains on the FPGA.

![Figure 6. Simplified scheme of video capture and processing on FPGA.](image)

| Problem to solve | Performance gain compared to: | Note |
|------------------|-------------------------------|------|
|                  | CPU  | GPU | |
| Speech recognition images – without taking into account delays | 43x  | 2x | GoogLeNet (unlimited number of training examples) |
| Image recognition 2 ms delay | -  | 8x | GoogLeNet (<1 ms), lower bound for CPU 5 ms. |

When comparing the performance of the Intel Agile FPGA family, and the GeForce GTX 2070 GPU, the graphs show that the implementation on the FPGA is superior to the implementation on the GPU. Figure 7 shows a graph comparing the performance of an FPGA and a GPU.
Figure 7. FPGA and GPU performance comparison graph.

Hardware acceleration using FPGAs provides good conditions for improving the performance of image processing algorithms compared to if all calculations were performed on the CPU or GPU. In the case of performing the same type of simple image processing operations, the acceleration when using an FPGA can reach more than 40 times (see Table 4). In turn, increasing the availability of specific resources in modern FPGAs, such as embedded multiplexers and registers, allows the FPGA to be adapted to meet many of the challenges and needs of modern image and video processing systems, in particular.

4. Conclusions
In the course of this work, the following results were obtained:
- an analysis of the market of innovative FPGA technologies was made. Based on the results of the analysis, a comparative table of the two most suitable solutions available on the market at the moment was compiled. Based on the data of the comparative table, the company Xilinx architecture Versal Premium series was selected;
- implemented a machine vision algorithm capable of recognizing handwritten Cyrillic characters based on the use of a pre-trained neural network AlexNet. The accuracy of the trained network is about 93%, which is a very good indicator;
- a comparative analysis of the performance of FPGA and GPU was carried out, during which it was found that the implementation on the FPGA can give a performance increase of about 40 times compared to the implementation on the GPU.

References
[1] Intel Agilex FPGAs and SoCs FPGA Family, URL: https://www.intel.ru/content/www/ru/ru/products/programmable/fpga/agilex.html
[2] Intel Cyclone 10 FPGAs Devices - INTEL FPGA, URL: https://www.intel.ru/content/www/ru/ru/products/programmable/fpga/cyclone-10.html
[3] Versal AI Core Series, URL: https://www.xilinx.com/products/silicon-devices/acap/versal-ai-core.html
[4] Tarasov I E 2020 Components and technologies 12 17-21 (in Russian)
[5] Dedegkhaev A G and Ryzhkov A A 2013 Universum. Technical sciences: electronic. 1(1) (in Russian)
[6] Adilov R M 2012 *Artificial intelligence systems. Module 3. Machine vision systems: training manual* (Penza: Publishing House “PenzaSTU”) p 32 (in Russian)

[7] Zoev I V, Ryzhova S E and Beresnev A P 2016 Youth and modern information technologies 45-46 (in Russian)

[8] Colbert I, Daly J and Kreutz-Delgado K 2021 *arXiv preprint arXiv:2102.00294v2*

[9] Aarrestad T, Loncar V and Pierini M 2021 *arXiv preprint arXiv:2101.05108v1*

[10] Watanabe H, Matsutani H 2012 *arXiv preprint arXiv:2012.15465*