



Copyright: Original content from this work may be used under the terms of the creative commons attributes 4.0 licence.

Review And Analysis Of Computer Vision Algorithms

Bakhtiyar Saidovich Rakhimov

Head Of The Department Of Biophysics And Information Technologies Of Urgench Branch Of Tashkent Medical Academy, Uzbekistan

Feroza Bakhtiyarovna Rakhimova

Assistant Professor Of The Department Of Biophysics And Information Technologies Of Urgench Medical Academy, Uzbekistan

Sabokhat Kabulovna Sobirova

Assistant Professor Of The Department Of Biophysics And Information Technologies Of Urgench Medical Academy, Uzbekistan

Furkat Odilbekovich Kuryazov

Assistant Professor Of The Department Of Biophysics And Information Technologies Of Urgench Medical Academy, Uzbekistan

Dilnoza Boltabaevna Abdirimova

Assistant Professor Of The Department Of Biophysics And Information Technologies Of Urgench Medical Academy, Uzbekistan

ABSTRACT

Computer vision as a scientific discipline refers to the theories and technologies for creating artificial systems that receive information from an image. Despite the fact that this discipline is quite young, its results have penetrated almost all areas of life. Computer vision is closely related to other practical fields like image processing, the input of which is two-dimensional images obtained from a camera or artificially created. This form of image transformation is aimed at noise suppression, filtering, color correction and image analysis, which allows you to directly obtain specific information from the processed image. This information may include searching for objects, keypoints, segments, and annexes;

KEYWORDS

Information technology, digital signal processing, mathematical models, network.

INTRODUCTION

Typical computer vision applications are:

- 1) Detection of objects in the image. Despite the fact that a person can easily select specific objects from an image, this task has not yet been completely solved for artificial systems. Most of the solutions are private in nature, based on the specific properties of the sought object, and, accordingly, are not suitable for searching for objects that do not have them. There are several universal algorithms for object detection (neural networks, Viola-Jones, etc.) that work slowly and have a serious detection error with insignificant deviations of objects from the desired ones, set during training, but, nevertheless, their simplified versions are widely used for small images. Therefore, an important task while maintaining the detection accuracy is to speed up the calculations;
- 2) Recognition of objects in the image. This task is a continuation of the previous one, the result of which is an array of areas where objects can be found. The purpose of this task is to determine the presence in these areas of a specific class of objects, which already has more specific features and, accordingly, can be better classified;
- 3) Identification of objects, the result of which can be a conclusion about the correspondence of the recognized object to a specific (unique) instance (for example, a fingerprint, the face of a specific person, car number, etc.). Separately, it is worth highlighting character recognition systems from this group, the identification accuracy of which affects the quality of the material being recognized;
- 4) Search for images in the database by content, based on the recognition of a specific class of objects. In this case, the performance of the artificial system plays a significant role in speeding up the search for images, therefore, the possibility of parallelizing the algorithms of this group of tasks is considered very important;
- 5) Reconstruction of a three-dimensional scene from a certain set of input images (video stream) allows you to determine the positions of objects and a source in three-dimensional space, used for the movement of robots, creating panoramic images, etc.;
- 6) Tracking moving objects in a video stream provides for the direct determination of the position of an object in space by changing its position on two-dimensional images while maintaining the characteristic features of the object. This task is very resource-intensive and must be performed in real time, so the main emphasis when creating algorithms for this area is made on their performance;
- 7) Image processing. This task area is intended for converting pixels of two-dimensional images and is of primary importance for other computer vision tasks. Almost all transformations are filter transformations, i.e. a set of operations is performed over each pixel of the image, depending on other pixels that are in close proximity to the desired one using special matrices.

OBJECTIVE STATEMENT

Obviously, not all computer vision algorithms can be parallelized on GPUs. Any artificial computer vision system, regardless of its area

of application, should include the following typical stages of work:

- 1) Image acquisition (photo or video filming);
- 2) Preliminary processing;
- 3) Highlighting characteristic features;
- 4) Detection or segmentation;
- 5) High-level processing.

Almost all stages can be realized with the help of parallel computer vision algorithms executed on modern parallel computing devices. Some of them can use data parallelism, which is advisable to use on the general-purpose GPUs discussed above. Consider the main groups of computer vision algorithms using data parallelism (this classification is a generalization of groups from):

- 1) Image transformation algorithms - input and output data are two-dimensional images, the coordinates of the output image element differ from the coordinates of the input element. These algorithms include: affine transformations, coordinate system transformations, etc.;
- 2) Filtering algorithms - input and output data are two-dimensional images. Each pixel of the output image is the result of an operation on a group of pixels in the input image that fall into a window of a certain size (filter). Filters can be represented by various mathematical devices: cellular automata, neural networks, functions, etc.;
- 3) Statistical algorithms - input data are two-dimensional images, output data are arrays

with statistical information. Each element of the original data can be used for statistical calculations if it meets the requirements of the algorithm. An example of such algorithms can be the calculation of histograms, Hook's transformation, grouping of elements, etc. Parallelization of such algorithms requires communication between processors to combine the accumulated statistical data;

- 4) Recursive algorithms - input and output data are two-dimensional images. Each element of the original image contributes to the formation of all image elements in the output. These algorithms include: calculation of the integral of the image, distance transformation, etc. This type of algorithms is very difficult to parallelize, since in most cases, the computation of a single output item requires the result of the previous input items.

The Viola-Jones method for recognizing objects in an image based on rectangular Haar features can be used as a superposition of algorithms of several groups. Despite the resource intensity of the method, it is widely used in various tasks. The method consists of three stages:

Creation of the integral of the image. An image integral is a two-dimensional array of sums of color intensities that has the same dimensions as the image itself, and the value in each cell is calculated using the following formula:

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (1)$$

This computation is classified as recursive, which makes it difficult to parallelize the algorithms that implement it. To parallelize the

algorithms for calculating the integral of the image, a division into two stages is used: horizontal prefix sums and vertical prefix sums. At each stage, for each row (or column), prefix sums of a one-dimensional data array are calculated using various algorithms: Horn's algorithm or Scan's algorithm (Hillis and Steele, 1986). These algorithms are data parallelized and implemented on GPUs. A combined algorithm can also be used depending on the amount of input data.

In some cases, it is required to find rotated integrals at a certain angle to the original image, which can be reduced to the problem of finding the integral of the image through the first implementation group for SIMD.

$$f(x) = \sum_{t=1}^T \alpha_t \cdot h_t(x), \quad (2)$$

where $h_t(x)$ - is a weak classifier;

α_t - is the weight of the given classifier in the cascade.

If the result of a strong classifier is less than a certain threshold value, then the object is considered detected. As weak classifiers, Viola-Jones uses Haar features - a weighted function of the brightness of rectangular regions inside a sliding window. An obvious way to parallelize this algorithm is to transfer the computations of each strong classifier to a separate process. However, when implementing a parallel algorithm on a graphics processor, there are problems of sparse hypotheses and idle processors associated with memory addressing patterns and simultaneous execution of bundles. These problems are

solved by scaling the image integral and pre-passing several weak classifiers.

Grouping of results. The purpose of this stage is to group the nearby possible areas of finding an object with their subsequent averaging to display the result of the method. Comparison is performed across all areas using a special union function. At the same time, a tree is being built, the root nodes of which will be the required search results. This algorithm has a complexity that with a large number of areas found at the second stage of the method can negatively affect performance in general. To parallelize this algorithm, you can transfer the calculation of the union function to the graphics processor.

The increased interest in the field of heterogeneous programming and the high availability of massively parallel devices in the form of GPUs lead to the creation of new parallel computing models that take into account the peculiarities of various parallel computing devices. In the course of the analysis, it was concluded that not a single abstract model of parallel computing known to date can fully describe the operation of the CPU-GPU system for various types of graphics processors. Nevertheless, a sufficient number of parallel algorithms have been created that can be implemented using this system. Therefore, the development of an abstract model of parallel computing on graphic processors and the corresponding abstract model of the CPU-GPU system can be considered an urgent task of the research. Obviously, this model will have the properties of the abstract parallel machines presented in the review, which, in turn, will allow transferring the algorithms for these models to the new system without significant changes.

It should be noted that in the CPU-GPU system, the memory hierarchy and the corresponding bandwidth of the data transmission channels will play a significant role. Therefore, in most cases, to improve the performance of algorithms, it will be advisable to cache data in various types of memory. Consequently, the study of different types of caching in the CPU-GPU system is an urgent task aimed at optimizing programs using parallel algorithms on GPUs.

CONCLUSIONS

The theoretical results of the study are supposed to be implemented and experimentally tested on various groups of computer vision algorithms, which in most cases use two-dimensional images as input and output data, converted into a one-dimensional N-element array. The results of these experiments will prove the consistency of the model and optimization methods. The Viola-Jones method should be separately noted as a resource-intensive general method for recognizing objects in an image using rectangular Haar features. This method contains several different computer vision algorithms and is a working example of 3 and 4 groups of algorithms, which were discussed above. In addition, this method is implemented on a CPU and is used in a complex for automatic functional testing of interactive graphics applications. Therefore, transferring the algorithms of this method to the graphics processor can increase the performance of the recognition module and, accordingly, the entire complex.

REFERENCES

1. Carl de Boor. A practical Guide to splines. Department of computer science University of Wisconsin – Madison, Madison, USA, 2001.
2. Ren-Hong Wang. Multivariate spline functions and their applications. Kuliver Academic Press, Netherlands, 2001.
3. Rakhimov B.S., Allabeganov O. R., Saidov A.B. PROCESSOR MEANS FOR THE SPECTRAL ANALYSIS OF MEDICAL SIGNALS ON THE OF POLYNOMIAL WALSH BASES EPRA International Journal of Research and Development (IJRD) Volume: 5 | Issue: 7 | July 2020 pages 10-11.
4. Rakhimov B.S., Ismoilov O.I., Ozodov R.O. Russian “Software and automation of forensic examination” METHODS OF SCIENCE Scientific and practical journal №11 / 2017 pages 28-30
5. Rakhimov B.S. Russian “Information technologies in medical education” METHODS OF SCIENCE Scientific and practical journal №12 / 2017 pages 25-27
6. Rakhimov Bakhtiyar Saidovich, Mekhmanov Mukhiddin Sultanovich, Bekchanov Bakhtiyar Gafurovich Parallel algorithms for the creation of medical database systems ICMSIT-II 2021: Metrological support of innovative technologies Conference materials
7. U. Pen. Application of Wavelets to Filtering of Noisy Data. In Wavelets: the Key to Intermittent Information? — Oxford University Press, 2000.
8. Ashok, V. Balakumaran, T. Gowrishankar, C. Vennila, A. Kumar, A Nirmal. The Fast Haar Wavelet Transform for Signal & Image Processing. International Journal of Computer Science and Information Security, IJCSIS, USA(2010), January 2010,

ISSN 1947 5500,
<http://sites.google.com/site/ijcsis/>

9. Bakanov V.M. Parallel computing: Textbook / V.M. Bakanov; MGUPI. Moscow, 2006.-124 p.
10. Boreskov, Kharlamov, Markovsky: Parallel computing on GPU. Architecture and software model of CUDA: Textbook, MGUPI. Moscow, 2012.-336 p.
11. Voevodin VV, Voevodin VI. V. Parallel computing / V.V. Voevodin, VI. V. Voevodin - SPb.: BHV-Petersburg, 2002. -- 608 p.: ill.
12. P. P. Kudryashov Algorithms for detecting a human face for solving applied problems of image analysis and processing: author. dis. Cand. tech. Sciences: 05.13.01. - M, 2007.
13. Tanenbaum E. Modern operating systems. 2nd ed. - SPb.: Peter, 2002. -- 1040 p.: ill.
14. Forsyth DA, Pons, J. Computer vision. Modern approach / D.A. Forsyth, J. Pons: Trans. from English - M.: Publishing house "Williams", 2004. - 928 p.: ill. - Parallel. tit. English
15. Frolov V. Solution of systems of linear algebraic equations by the preconditioning method on graphic processor devices