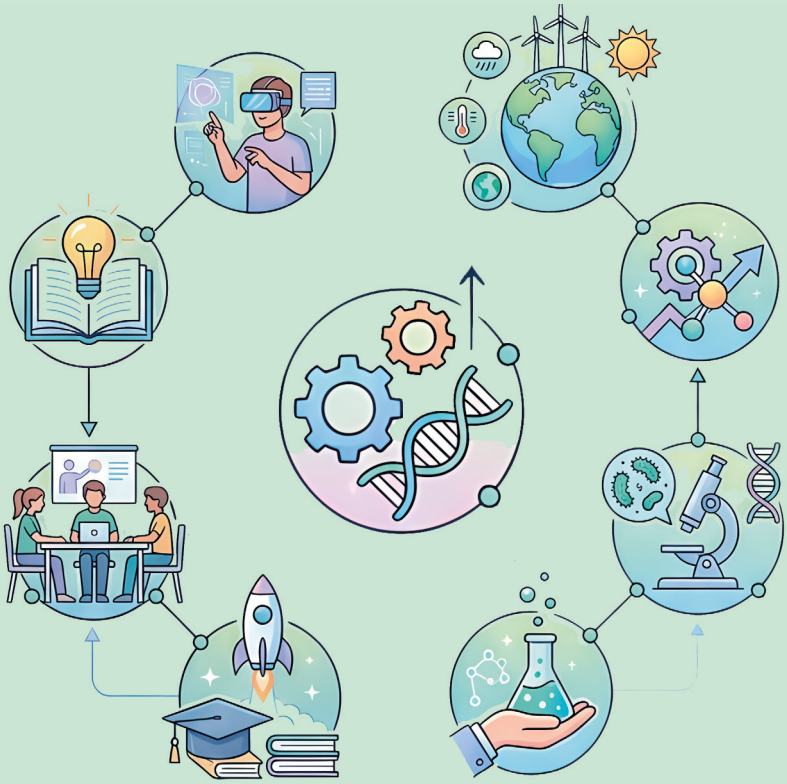
ISP
international science project

Sherman Oaks California (USA), 2026



INTELLECTUAL HORIZONS:

AN INTERDISCIPLINARY SYNTHESIS OF SCIENTIFIC KNOWLEDGE AND EDUCATIONAL INNOVATIONS

COLLECTIVE MONOGRAPH

INTELLECTUAL HORIZONS: AN INTERDISCIPLINARY SYNTHESIS OF SCIENTIFIC KNOWLEDGE AND EDUCATIONAL INNOVATIONS

Compiled by
VIKTOR SHPAK

Chairman of the Editorial Board
STANISLAV TABACHNIKOV

GS Publishing Services
Sherman Oaks
2026

The collective monograph is a scientific and practical publication that contains scientific articles by doctors and candidates of sciences, doctors of philosophy and art, graduate students, students, researchers and practitioners from European and other countries. The articles contain research that reflects current processes and trends in world science.

Text Copyright © 2026 by the Publisher «GS Publishing Services» and authors.
Illustrations © 2026 by the Publisher «GS Publishing Services» and authors.
Cover design © 2026 Publisher «GS Publishing Services».

Authors: Oksana Antolik, Olena Berezinska, Oleksii Bobko, Halyna Boikivska, Anastasia Boychuk, Alla Cherep, Oleksandr Cherep, Larysa Duda, Svitlana Faizulina, Dmytro Fedenko, Anatoliy Konokh, Andriy Konokh, Olena Konokh, Nina Korbozerova, Petro Krul, Oleksandr Lesko, Valentyna Moyseyenko, Anton Nosov, Oleksandr Novoseltsev, Andrii Nykula, Viktoriia Ogloblina, Lyudmila Oleynikova, Oleksandr Prytula, Natalia Punchenko, Oleksandr Romanyuk, Iryna Satsyuk, Anatolii Shamanskyi, Ivan Shcherbyna, Tetiana Skliaruk, Oleksii Stakhov, Dmytro Svetlyi, Ilona Tkachuk, Vira Tymoshenko, Valentyna Voronkova.

All rights reserved. No part of this publication may be reproduced, distributed or transmitted in any form or by any means, or stored in a database or search engine without the prior written permission of the publisher. The authors are responsible for the content and reliability of their articles. Citation or other use of the monograph is possible only with reference to the publication.

Publisher «GS Publishing Services»
15137 Magnolia Blvd, # D,
Sherman Oaks, CA 91403, USA.

ISBN 979-8-9935428-7-4
DOI: 10.51587/9798-9935-42874-2026-029

Reviewers:
Hanna Kozhyna, Doctor of Medical Sciences, Professor, Head of the Department of Psychiatry, Narcology, Medical Psychology, and Social Work at Kharkiv National Medical University.
Volodymyr Verbytskyi, Doctor of Pedagogical Sciences, Professor, Director of the National Ecological and Naturalistic Center for Student Youth of the Ministry of Education and Science of Ukraine.

Scientific editors-reviewers:
S. Bobrovnyk, Yu. Bondar, A. Cherep, P. Glukhovskiy, P. Hovorov,
Yu. Kuznietsov, V. Lazurenko, V. Moyseyenko, L. Omelianchyk, Zh. Virna

The monograph is recommended for publication by the Academic Council of the Institute of Adult Education of the PO «National Academy of Sciences of Higher Education of Ukraine» (Protocol No. 6 dated September 23, 2026).

Intellectual Horizons: an Interdisciplinary Synthesis of Scientific Knowledge and Educational Innovations : collective monograph / Compiled by V. Shpak; Chairman of the Editorial Board S. Tabachnikov. Sherman Oaks, California : GS Publishing Services, 2026. 221 p.

Available at: DOI: 10.51587/9798-9935-42874-2026-029

CONTENT

COMPUTER SCIENCE

Oleksii BOBKO

METHODS OF CONTAINERIZING THE RENDERING OF THREE-DIMENSIONAL GRAPHIC IMAGES 6

Oleksandr ROMANYUK
Oleksii STAKHOV

METHODS FOR ACCELERATING THE RENDERING OF THREE-DIMENSIONAL SCENES 14

CULTUROLOGY

КРУЛЬ Петро Франкович,
ДУДА Лариса Ігорівна

ЦЕХОВА МУЗИКА МАГНАТСЬКИХ ТА КОЗАЦЬКО-СТАРШИНСЬКИХ МАЄТКІВ XVII – ПОЧАТКУ XIX СТОЛІТЬ 44

ECONOMIC SCIENCES

БОЙКІВСЬКА Галина Миколаївна

ТРАНСФОРМАЦІЯ ПІДХОДІВ ДО РОЗВИТКУ ТАЛАНТІВ В ОРГАНІЗАЦІЇ: ВІД УПРАВЛІННЯ ТАЛАНТАМИ ДО ФОРМУВАННЯ ЕКОСИСТЕМНОЇ МОДЕЛІ 54

ВОРОНКОВА Валентина Григорівна,
НІКІТЕНКО Віталіна Олександрівна,
ОГЛОБЛІНА Вікторія Олександрівна,
ПУНЧЕНКО Наталія Олегівна

ЦИФРОВА ТРАНСФОРМАЦІЯ МЕНЕДЖМЕНТУ В УМОВАХ НЕВИЗНАЧЕНОСТІ ТА ТУРБУЛЕНТНОСТІ: ВІД INDUSTRY 4.0 ДО MANAGEMENT 5.0 61

Oleksandr LESKO,
Anatolii SHAMANSKYI

INTELLIGENT CONTROL SYSTEMS FOR MANUFACTURING PROCESSES 76

СКЛЯРУК Тетяна Василівна

МІЖДИСЦИПЛІНАРНИЙ ПІДХІД ДО ФОРМУВАННЯ УПРАВЛІНСЬКИХ, ЦИФРОВИХ І СОЦІАЛЬНИХ КОМПЕТЕНТНОСТЕЙ МАЙБУТНІХ HR-ФАХІВЦІВ 86

ФЕДЕНКО Дмитро Олександрович

ПІДХОДИ ДО ВИКОРИСТАННЯ ВІЗУАЛЬНИХ КОНФІГУРАТОРІВ ПРИ ІМПЛЕМЕНТАЦІЇ ПОКАЗНИКІВ ІНТЕГРОВАНОГО ЗВІТУ ТА ЗВІТУ ЗІ СТАЛОСТІ..... 98

ЧЕРЕП Алла Василівна,
ОЛЕЙНІКОВА Людмила Григорівна,
НОСОВ Антон Дмитрович,
ЩЕРБИНА Іван Сергійович

СТРАТЕГІЯ СТАЛОГО РОЗВИТКУ ТА ЕКОЛОГІЧНА СТІЙКІСТЬ 121

Череп Олександр Григорович,
ФАЙЗУЛІНА Світлана Айгалівна,
СВІТЛИЙ Дмитро Олександрович,
БОЙЧУК Анастасія Сергіївна

ДОСВІД КРАЇН ЄС ЩОДО ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ МАРКЕТИНГОВОЇ ДІЯЛЬНОСТІ ЦИФРОВИХ ТЕХНОЛОГІЙ 131

ЧЕРЕП Алла Василівна,
АНТОЛІК Оксана Миколаївна

ЦИФРОВІЗАЦІЯ ГРОМАДСЬКИХ ОРГАНІЗАЦІЙ ВЕЛИКОБРИТАНІЇ ТА УКРАЇНИ: ТЕНДЕНЦІЇ ТА ВИКЛИКИ 138

LINGUISTICS

БЕРЕЗІНСЬКА Олена Володимирівна

ІМЕРСИВНІ ТЕХНОЛОГІЇ ТА КЕЙС-МЕТОД У ФОРМУВАННІ ПРОФЕСІЙНО ОРІЄНТОВАНОЇ ІНШОМОВНОЇ КОМПЕТЕНТНОСТІ МАЙБУТНІХ ФАХІВЦІВ (НА ПРИКЛАДІ АНАЛІЗУ РИНКУ *BLEISURE-ТУРИЗМУ*) 145

КОРБОЗЕРОВА Ніна Миколаївна

МОВНА СИТУАЦІЯ ПІРЕНЕЇВ ДОРИМСЬКОГО ПЕРІОДУ 162

MEDICINE

МОЙСЕЄНКО Валентина Олексіївна,
НИКУЛА Андрій Тарасович

РОЛЬ ЛАКТОБАКТЕРІЙ У ПІДТРИМЦІ ГОМЕОСТАЗУ ОРГАНІЗМУ 167

PEDAGOGICAL SCIENCES

ТКАЧУК Ілона Львівна

КОГНІТИВНА МАТРИЦЯ СПРИЙНЯТТЯ: ЧОМУ КЛАСИЧНИЙ РИСУНОК І ЖИВОПИС ЗАЛИШАЮТЬСЯ НЕЗАМІННИМИ В ЕПОХУ DIGITAL ART 171

ПРИТУЛА Олександр Леонтійович,
КОНОХ Анатолій Петрович,
КОНОХ Олена Євгеніївна,
КОНОХ Андрій Анатолійович

ФІЛОСОФСЬКО-ПЕДАГОГІЧНИЙ ПОТЕНЦІАЛ КОЗАЦЬКОЇ СПАДЩИНИ У ФОРМУВАННІ ГОТОВНОСТІ МАЙБУТНІХ ФАХІВЦІВ ФІЗИЧНОЇ КУЛЬТУРИ І СПОРТУ ДО ЗАХИСТУ БАТЬКІВЩИНИ 176

POLITICAL AND LEGAL SCIENCES

ТИМОШЕНКО Віра Іванівна

ІСТОРИЧНИЙ ПОГЛЯД НА КРИМІНАЛІСТИЧНУ ІДЕНТИФІКАЦІЮ ОСОБИ ЗА БІОМЕТРИЧНИМИ ДАНИМИ 190

TECHNICAL SCIENCES

Oleksandr NOVOSELTSEV

RELIEF TEXTURING METHODS BASED ON REPRESENTING FREE SPACE WITH GEOMETRIC PRIMITIVES IN TWO-DIMENSIONAL TEXTURES 206

Oleksandr ROMANYUK,
Iryna SATSYUK

ANALYSIS OF DEVELOPMENT DIRECTIONS OF VIDEO CARD ARCHITECTURES 213

slopes and irregular stone at frontal and grazing angles. Measurements should cover intersection error, missed first hits, temporal stability, GPU time, texture fetches, memory and preprocessing cost. Reference intersections must use the same interpolation rule. These directions remain research questions until comparisons demonstrate an improvement.

Compact geometric descriptions of free space can preserve surface detail without dense geometry or large volumetric textures. Their development should connect reliable spatial bounds with precise local intersection, preserving both speed and correct visibility.

DOI: 10.51587/9798-9935-42874-2026-029-206-213

Oleksandr ROMANYUK,
Doctor of Technical Sciences, Professor,
Vinnytsia National Technical University
ORCID iD: 0000-0002-2245-3364

Iryna SATSYUK,
Student,
Vinnytsia National Technical University
ORCID iD: 0009-0005-0130-6233
Ukraine

ANALYSIS OF DEVELOPMENT DIRECTIONS OF VIDEO CARD ARCHITECTURES

A modern video card is a complex specialized computing system, the functional purpose of which has long gone beyond the classical formation of two-dimensional and three-dimensional graphics. It combines a graphics processor, local video memory, a multi-level caching system, memory controllers, video encoding and decoding units, display controllers, data exchange interfaces with the central processor, a power management system and a set of specialized computing modules. The main element of a video card remains the GPU, however, it is advisable to consider a modern GPU not as a separate graphics accelerator, but as a heterogeneous multiprocessor system optimized for massive parallelism. This approach corresponds to the general trend in the development of computer architectures, in which, along

with universal processors, vector, graphic and object-oriented accelerators are becoming increasingly important[1].

The evolution of graphics card architecture is characterized by a gradual transition from a rigidly fixed graphics pipeline to a programmable structure with a large number of identical execution elements. The first 3D accelerators contained separate hardware blocks for geometric transformations, rasterization, texturing, and color calculations. This organization was effective for a certain set of operations, but did not adapt well to changes in the structure of the graphics load. The emergence of programmable vertex and pixel shaders changed the approach to building GPUs: some of the functions that were previously implemented by fixed logic became possible to specify programmatically. The next step was the formation of unified shader blocks capable of executing different types of programs depending on the current stage of the graphics pipeline.

The basic direction of GPU development remains the increase in parallelism. Graphics tasks naturally allow simultaneous processing of a large number of vertices, fragments, pixels, texture samples and rays. That is why GPUs are focused on high throughput, and not on the minimum delay of execution of one sequential thread. In practical implementation, SIMD- and SIMT-like principles are used, according to which one instruction or one sequence of instructions is executed over a group of data elements or threads. A large number of simultaneously active threads also allows to hide memory access delays: while one group is waiting for the loading to complete, the scheduler can transfer execution resources to another group. Such an organization is one of the reasons for the high efficiency of GPUs not only in graphics, but also in numerical modeling and machine learning.

At the same time, a simple increase in the number of computational elements does not provide a proportional increase in performance. At a high level of parallelism, performance begins to be determined by memory limitations, the bandwidth of internal connections, the efficiency of thread dispatching, the size of the register file, and the nature of branching in the program. If neighboring threads execute different branches of a conditional operator, part of the computational resources may be temporarily unused. Similar losses

1 Hennessy JL, Patterson DA, Kozyrakis C. Computer Architecture: A Quantitative Approach. 7th ed. Morgan Kaufmann, 2025. 936 p.

occur with irregular access to memory or excessive use of registers. Therefore, modern architectures develop not only quantitative characteristics, but also mechanisms for increasing the actual load of computational blocks: schedulers, work regrouping mechanisms, local memory operation, and means of hiding delays are being improved.

A striking example of the transition to a heterogeneous organization is the NVIDIA RTX Blackwell architecture. Its streaming multiprocessor combines universal CUDA cores, texture modules, local cache, register resources, Tensor Cores and other functional blocks, and at the GPU level, specialized RT Cores are additionally used. The architecture is focused not only on traditional rasterization rendering, but also on neural methods of image formation, reduced-precision calculations and hardware-accelerated ray tracing. The Blackwell technical description highlights neural shading, fourth-generation RT Cores, fifth-generation Tensor Cores, FP4 support and new power management mechanisms among the key areas[2]. This shows that a modern GPU is turning into a system in which different types of blocks are specialized for different classes of operations.

Specialization of hardware modules is a natural response to the increasing complexity of graphics algorithms. A universal shader unit can theoretically implement many functions programmatically, but such an implementation is not always optimal in terms of power consumption and the number of clock cycles. If a certain operation is performed extremely often and has a stable structure, it is advisable to create a separate hardware accelerator for it. It is on this principle that texture modules, rasterization units, specialized ray tracing units, matrix accelerators, video encoders and decoders appeared in the GPU. As a result, the area of the crystal is divided between universal and object-oriented resources. The architect's task is to choose a ratio that provides the highest efficiency for the target class of applications.

One of the most notable areas of specialization has been hardware ray tracing. Classical rasterization remains a very effective way to build a frame, but the reproduction of physically plausible reflections, refractions, soft shadows and complex global illumination requires additional methods. Ray tracing allows you to directly model the propagation of light, but requires

2 NVIDIA Corporation. NVIDIA RTX Blackwell GPU Architecture. Version 1.1. 2025. 57 p. URL : https://images.nvidia.com/aem-dam/Solutions/geforce/blackwell/nvidia-rtx-blackwell-gpu-architecture.pdf ((Accessed 19.09.2026).

performing a large number of operations to find intersections of rays with geometric objects and bypass spatial acceleration structures. Performing all these operations with universal shader cores leads to significant resource consumption. Therefore, modern video cards contain specialized blocks that accelerate the most characteristic operations of the ray algorithm and reduce the load on the main shader array.

The spread of hardware ray tracing has formed a hybrid rendering model in which rasterization and ray tracing methods are used together. Rasterization provides high processing speed of the main geometry, while ray tracing is used for those effects where it gives the greatest gain in quality. This approach allows you to achieve a compromise between realism and frame rate. Further development of RT-blocks is associated not so much with a mechanical increase in their number, but with increasing the efficiency of working with complex geometry, dynamic scene structures and large arrays of rays. Of particular importance are mechanisms that reduce the divergence of flows and increase the locality of data access when bypassing acceleration structures.

In parallel with ray tracing, the integration of artificial intelligence tools is rapidly developing. Matrix calculations, which are the basis of modern neural networks, have high regularity and scale well on parallel hardware. The inclusion of specialized matrix blocks allows you to perform matrix multiplication and accumulation of results much faster compared to universal arithmetic devices. Initially, such modules were considered mainly as a means of accelerating machine learning tasks, but now they are increasingly integrated directly into the graphics pipeline. Neural models are used for frame scaling, detail reconstruction, smoothing, noise reduction of ray tracing results, generation of intermediate frames and restoration of visual information that does not necessarily have to be calculated in the traditional way at full resolution.

The integration of neural algorithms has led to the development of support for reduced-precision numerical formats. For traditional graphics, 32-bit and 16-bit floating-point formats are widely used, but for many neural inference operations, such precision is overkill. The transition to FP8, FP4, or related integer formats allows you to perform more operations in the same hardware area, reduce the amount of data transfer, and reduce power consumption. Importantly, reduced precision is used selectively: critical stages can operate

with higher precision, while massive matrix operations are performed in more compact formats. As a result, a modern GPU becomes a processor with flexible arithmetic precision, where the specific format is determined by the requirements of the algorithm.

AMD RDNA 4 demonstrates a similar trend. This generation of architecture improves the computational units, specialized ray tracing accelerators and AI accelerators, and the instruction set is focused on combining graphics and matrix loads. The official RDNA 4 reference describes the execution environment, organization of the program state and the capabilities of the new generation of computational units[3]. To analyze development directions, it is important not so much to compare the marketing indicators of individual models, but to note the general architectural pattern: universal shader resources are supplemented by specialized AI and RT modules, and the efficiency of the graphics processor is determined by the level of their integration into a single pipeline.

Another important trend is observed in the Intel Xe family. The Xe architecture covers solutions for integrated, discrete and high-performance graphics, and the Xe2 generation is used in high-performance graphics configurations. The Xe approach combines vector execution resources, matrix mechanisms and multi-level memory. The official oneAPI GPU Optimization Guide emphasizes that the performance of different Xe microarchitectures is determined by the combination of computational capabilities, cache sizes, supported data types, manufacturing technology and frequency characteristics[4]. So, in this case, we can see a transition from evaluating the GPU by a single parameter to a multi-factor analysis of the entire structure.

To summarize the main directions, it is advisable to compare the characteristic architectural accents of modern solutions. As can be seen from Table 1, in all three approaches the massively parallel universal computing array is preserved, but an increasingly complex system of specialized functional blocks is formed around it. This allows the GPU to be adapted to tasks that were previously either performed programmatically at great expense, or were not considered as part of the graphics pipeline at all.

3 Advanced Micro Devices, Inc. "RDNA 4" Instruction Set Architecture: Reference Guide. 2025. URL : https://docs.amd.com/v/u/en-US/rdna4-instruction-set-architecture (09/19/26).

4 Intel Corporation. oneAPI GPU Optimization Guide. Intel Xe GPU Architecture. Version 2025.2. 2025. URL : https://www.intel.com/content/www/us/en/docs/oneapi/optimization-guide-gpu/2025-2/intel-xe-gpu-architecture.html (Accessed 19.09.2026).

Table 1

Typical architectural trends of modern GPUs

Source: summarized by the author

Architectural direction	NVIDIA RTX Blackwell	AMD RDNA 4	Intel Xe/Xe2
Basic parallel structure	Streaming Multiprocessor	Compute Unit	Xe-core / execution resources
Ray tracing	RT Cores	Raytracing Accelerators	Hardware RT blocks
Matrix / AI computing	Tensor Cores	AI Accelerators	Matrix executive resources
Key development direction	Neural rendering, mixed precision	Combining graphics, RT and AI	Xe scalability and different GPU classes

The comparison shows that the development of graphics card architecture does not boil down to a single scheme, but the directions of evolution of different manufacturers coincide in essence. First, the universal parallel basis necessary for shader and general computing tasks is preserved. Second, the set of specialized modules is expanding. Third, multi-level memory, data locality and scheduling mechanisms are becoming increasingly important. Fourth, the role of software-hardware interaction is increasing: a modern GPU function is practically worthless without driver, compiler and API support. That is why new generations of graphics cards should be analyzed as complex platforms, and not as a set of separate computing units.

One of the most difficult architectural problems remains the memory subsystem. Increasing the number of execution elements makes sense only when they receive data quickly enough. Modern rendering simultaneously uses large texture arrays, geometry buffers, depth maps, ray tracing structures, neural network parameters and intermediate post-processing results. All this creates significant traffic between computational units and video memory. Therefore, the development of GPUs is accompanied by an increase in GDDR bandwidth, improvement of memory controllers, expansion of caches and introduction of hardware compression. In professional accelerators, a similar role is played by HBM, which provides a very wide data access bus.

Cache memory improvements are not only important for performance, but also for energy efficiency. Accessing a local register or cache requires significantly

less energy than accessing external memory. Therefore, increasing the effective capacity of the cache, optimizing its usage policy, and reusing already loaded data can provide significant gains without a proportional increase in frequency or bus width. A multi-level hierarchy that includes registers, local or shared memory, a first-level cache, specialized caches, and a second-level cache becomes one of the central elements of the architecture. Its quality is especially important for ray tracing and neural algorithms, where the working data sets can be very large.

A separate direction is the development of the geometric part of the pipeline. Realistic scenes contain more and more geometric primitives, and the traditional formation of a large number of separate drawing commands creates a load on the CPU. Therefore, modern graphics APIs and GPUs are focused on GPU-driven rendering, mesh shaders, clustered geometry representation, and hardware or software rejection of invisible elements. Some of the decisions that were previously made by the central processor are transferred closer to the graphic data. This reduces the number of exchanges between the CPU and GPU and allows processing more complex scenes. Coordination of the geometric stage with ray tracing is especially important, since increasing detail directly increases the requirements for spatial search structures and memory volumes.

Reducing process standards has long been the main way to increase the GPU transistor budget. However, further scaling of semiconductor production is becoming increasingly difficult and expensive. Large monolithic dies are expensive, more difficult to manufacture, and more dependent on the yield of usable products. One answer to this is the chiplet organization, in which the functions of a large processor are distributed across multiple dies. This approach is already actively used in high-performance computing accelerators and can potentially be more widely used in graphics systems. Its advantages are better scalability and the ability to combine dies manufactured using different process standards.

A critical limitation of modern graphics cards is power consumption. Increasing the frequency, number of execution units and memory bandwidth increases the thermal load and complicates the power system. Therefore, performance per watt becomes one of the key criteria for architectural quality. Dynamic frequency and voltage regulation, clock gating, power gating, local

shutdown of unused units and fast redistribution of the energy budget between subsystems are used. Specialization also increases energy efficiency: an operation implemented by a separate hardware accelerator often requires less energy than an equivalent sequence of universal instructions. A similar effect is provided by reduced precision and reduced memory traffic.

Another direction of development is to optimize the exchange between the central and graphics processors and increase the efficiency of performing heterogeneous tasks on a single GPU. A discrete graphics card has local video memory, so large data sets often need to be moved between system memory and VRAM via PCI Express. This creates delays and can limit the performance of tasks with frequent exchange. Modern systems develop unified addressing models, hardware support for virtual memory, and mechanisms for direct access to large data areas. At the same time, the role of hardware multitasking is growing: graphics shaders, computational cores, neural operations, and multimedia blocks can work in parallel, and schedulers must distribute resources between them without significant downtime. Therefore, the exchange speed, support for asynchronous queues, and the quality of dispatching become as important characteristics as the number of execution blocks.

Hardware complexity makes software-hardware coevolution increasingly important. Real-world GPU performance is determined not only by the microarchitecture, but also by the compiler, driver, graphics API, and application structure. The compiler must efficiently allocate registers, form thread groups, optimize memory access, and use specialized instructions. Graphics APIs are gradually incorporating ray tracing, mesh shaders, variable shader rates, and other capabilities of new generations of GPUs. Without appropriate software support, a hardware feature provides no practical benefit, so the development of the architecture and software ecosystem effectively becomes a single process.

A separate promising direction is associated with the transition from the monolithic organization of large graphics processors to a modular layout. As the area of the crystal increases, the yield of usable products deteriorates and the cost of manufacturing increases, so the distribution of functions between several crystals or chiplets becomes technologically attractive. For GPUs, such a transition is more difficult than for central processors, since the graphics pipeline requires extremely intensive exchange between executive units, cache and memory controllers. An effective multi-chip video card must hide inter-chip delays, provide high internal

interface bandwidth and minimize data duplication. Therefore, the development of chiplet approaches should be considered together with the development of large caches, hardware task distribution and new packaging technologies. In the future, modularity may allow computing, memory and input-output resources to be independently scaled, forming various GPU configurations based on common structural components.

Thus, the analysis of modern architectures shows that the development of video cards occurs simultaneously in several interconnected directions. The massively parallel structure of the GPU continues to scale, but the main increase in efficiency is increasingly provided not by a simple increase in the number of executive elements, but by specialization, flexible accuracy of calculations, improvement of the memory hierarchy, hardware ray tracing, matrix accelerators and intelligent resource management. At the same time, the role of energy efficiency, new layout technologies, software-hardware interaction and scaling to multiple devices is growing. It is the complexity of these changes that determines the current stage of GPU development.

The most significant trend is the formation of a heterogeneous architecture, focused not on a single rendering algorithm, but on the joint execution of rasterization, ray tracing, neural processing and general parallel computing. In such a system, performance is determined by the ability to quickly move data, load heterogeneous execution modules and select the most efficient type of hardware for each stage. Subsequent generations of graphics cards are likely to develop this heterogeneous model.

DOI: 10.51587/9798-9935-42874-2026-029-213-221