An Expert System for Managing the Render Farms in Cloud Data Centers

Auday Al-Dulaimy
Mälardalen University
Dalarna University
Sweden
auday.aldulaimy@mdu.se

Karam Turki
Gilgamesh Studio
Jordan
karam.turki@gilgameshstudio.com

Thomas Nolte
Mälardalen University
Sweden
thomas.nolte@mdu.se

Alessandro V. Papadopoulos
Mälardalen University
Sweden
alessandro.papadopoulos@mdu.se

***Abstract*—The users of cloud services prioritize cost and performance, but they increasingly demand sustainable practices. Sustainability is no longer a choice for businesses but a strategic imperative that shapes global industries. This paper presents a new system for utilizing the render farms in cloud data centers. The system aims to reduce energy consumption and costs in cloud data centers while maintaining a specific level of performance, particularly when rendering images and videos. The system can be described as a cloud-based expert system that offers rendering as a service, while considering user preferences for performance, cost, and energy efficiency. The system reads different scene rendering parameters and accordingly chooses the most suitable GPUs that fit the user's requirements. In other words, the system inputs are the scene complexity and user preferences. The output is the optimal GPU for rendering. Scene complexity is determined based on several parameters, such as the number of frames and polygons, resulting in one scene-related value. The user preferences are also normalized to a preferences-related value. Then, these values are aggregated to determine the optimal available GPU to render the scene at the lowest cost, minimum possible energy consumption, and highest performance.**

***Index Terms*—Cloud computing, data centers, Render farms, Rendering-as-a-Service, Performance, Cost, Energy efficiency.**

I. Introduction

Computer graphics, 3D animation, and virtual and augmented reality experiences are multi-stage processes that can be complex, costly, and time-consuming. Each stage involves a range of tasks, but the rendering stage is the most complex and resource-intensive stage in such processes [1]. Rendering is the process of generating an image or sequence of images from scenes. This involves complex calculations that simulate the behavior of light and materials in a virtual environment. The resulting images must be of high quality and resolution to meet consumers' demands. However, creating such high-quality output requires significant computational power, which can be very expensive. As a result, animation studios and other involved users often face performance and economic issues when it comes to rendering, especially when they need to produce high-resolution output, such as 8K or higher. Investing in expensive hardware, such as high-end CPUs, GPUs, and storage systems, is the solution to get this level of quality, but it can significantly impact profitability. Thus, finding other solutions to optimize rendering performance and reduce costs is important. This can involve a range of strategies, such as using cloud-based rendering services.

The cloud computing model offers a wide range of services [2]. Switching to cloud-based rendering doesn't require expensive hardware or daily maintenance costs, but at the same time, it is still costly, time-consuming, and results in an environmental footprint. Moreover, especially when it comes to rendering, cloud providers' policy is to offer resources like GPUs with fixed specifications and charge users for usage based on fixed prices. This policy is not the best option for many users and needs to be more user-oriented. At the same time, any policy must consider other factors, like sustainability.

Cloud data centers' performance and sustainability can potentially revolutionize the IT industry. Cloud service providers can demonstrate their commitment to mitigating environmental impact and fostering social responsibility by prioritizing both performance and sustainability, in addition to other user preferences. Using advanced technologies and concepts, such as Artificial Intelligence (AI), to ensure efficient performance and sustainability can greatly benefit data centers. In this paper, we propose a system that enables users to specify their preferences (performance, cost, and energy) when submitting their images or scenes for rendering, and accordingly, the cloud provider specifies and allocates a GPU that best suits rendering the scene. Both user preferences and provider available resources are considered in a system with a defined rule engine mechanism to select the best available GPU for each scene. The system helps improve performance in cloud data centers that deliver services like Rendering-as-a-Service (RaaS).

The main contribution of this paper is presenting a system that considers different users' preferences while they request rendering services from cloud providers. To the best of our knowledge, this is the first system that allows users to specify preferences other than cost when selecting GPUs to perform rendering tasks.

The rest of the paper is organized as follows. Section II gives background information on the topic. Section III describes the proposed system in detail. Section IV evaluates the results of the system. Section V discusses previous related systems and works, while Section VI concludes the paper.

II. Background

In This section, we will introduce some of the principles background and explain some paper-related subjects.

A. GPU rendering

When compared to the CPUs, the GPUs can complete different tasks much faster. This is due to their ability to break tasks down into smaller components and finish them in parallel [3], thanks to hundreds of cores and many more threads they have. In rendering, using GPUs can be more efficient than using CPUs. Each scene comprises frames with several components or properties, such as polygons. The cores of each GPU process different frames, such that the threads in each core handle the components.

B. Rendering in cloud data centers

Cloud-based GPU rendering is increasingly becoming the preferred option for studios and other users, and it's all thanks to the numerous benefits it offers over traditional in-house workstations. Among these benefits are scalability, which allows for the rendering of larger and more complex scenes; accessibility, which enables users to work from anywhere and on any device; reduced hardware requirements, which saves costs; collaboration, which enables multiple users to work on the same project simultaneously; and sharing, which allows for easy sharing and distribution of files. The cloud render farm utilizes up to thousands of GPU nodes, each having many GPUs. The rendering is provided as a service via the Internet, and is likely to be accessed or viewed through online platforms.

C. Expert systems

Expert systems refer to computer systems that are designed to imitate the decision-making ability of a human expert. These systems are a part of artificial intelligence and use the knowledge of an expert in a specific field to create an automatic decision-making system that can operate in a real-time fashion [4]. To create an expert system, we need a knowledge base and inference rules used to make decisions. The decisions are taken based on knowledge obtained from experts and on users' inputs. The inference rules are developed collaboratively between the system designers and the experts in the field [5]. The main components of any expert system include the *Knowledge Base* to store the expert's knowledge, and the *Inference Engine* to store the concept of the expert's reasoning [6].

D. Scene complexity

Many components or parameters affect the scene complexity, which in turn affects the amount of resources required for rendering and has impacts on rendering results. These parameters can be summarized as follows: Number of Polygons (Poly), Vertex Count (Ver), Object Count (Obj), Texture Resolution (Res), Number of Light Sources (Lit), Number of Materials (Mat), Animation Complexity (Ani), Environmental Effects (Eff), and Number of Frames (Fra).

These parameters can be extracted and read manually in computer graphics software tool sets like Blender and in other computer graphics applications software like Maya. However, it is also possible to read these parameters automatically by developing a code.

E. GPU types used in the render farm

Nowadays, render farms use various types of advanced GPUs, all of which work efficiently. In this work, we assign hypothetical names to the specifications of different brands and models. In this paper, when mentioning the performance of a GPU, we are specifically referring to its speed performance, which is determined by the processor's speed, the number of cores in the processor, and the working memory associated with the GPU. Additionally, the cost or price of using the GPU is the amount required for operating and utilizing this resource per unit of time. Lastly, the energy consumption of the GPU refers to the amount of energy consumed during the utilization of the GPU, and it is measured in Kwh.

III. Methodology

This work aims to develop a cloud expert system for rendering (complex or simple) scenes, considering two perspectives: users' preferences for performance, cost, and energy consumption, and cloud providers' render farm configurations.

A. System model

The system model of this work simulates the render farms in a cloud data center, which utilizes thousands of GPU nodes, each having many GPUs, that share a huge pool of working memory named cloud memory. These nodes are connected to management consul and storage via a high-speed network.

In our work, we introduce ten hypothetical GPU models that are based on real GPUs currently in the market. We attached a rank for performance, cost of use, and energy consumption with each GPU model.

B. System interface

In the cloud computing model, rendering is available as a service over the Internet [1], and is expected to be accessed or viewed through various online platforms, like websites. We designed a web-based interface to request rendering, where users can optimize three tuning parameters as follows:

- Users have the option to set the desired performance level within a range of 1 to 10.
- Users are able to choose the desired level of cost depending on their budget within a range of 1 to 10.
- Users can set their preferred energy consumption level within the range of 1 to 10.

These values are aggregated with other values extracted from the scenes that reflect their complexity, and then, based on all values, the best-fit GPU is chosen. We will focus only on the three components that are most relevant to this work: the Knowledge Base, Rules Inference Engine, and User Interface.

C. Users preferences

Normalization transforms values or features to be on a similar scale, improving any model's performance and training stability. In this work, all user inputs are normalized to a uniform scale to be combined and compared. So, each input is transformed to a range from 0.1 to 1. The Performance (P), Cost (C), and Energy (E) are converted as follows:

$$P^N = \frac{P}{10} \tag{1}$$

$$C^N = \frac{C}{10} \tag{2}$$

$$E^N = \frac{E}{10} \tag{3}$$

where:
P^N is the normalized performance (rendering speed).
C^N is the normalized cost (rendering cost).
E^N is the normalized energy (rendering consumed energy).

In addition to normalizing the users' preferences, it is possible to assign a weighting value to each normalized input (wP^N, wC^N, wE^N) to reflect its importance, such that:

$$wP^N + wC^N + wE^N = 1 \tag{4}$$

In this work, we assumed that all inputs have equal importance, as follows:

$$wP^N = wC^N = wE^N = \frac{1}{3} \tag{5}$$

However, the cloud providers can change or tune these weights. The score for each weighted preference is calculated as follows:

$$swP^N = P^N \times wP^N \tag{6}$$

$$swC^N = C^N \times wC^N \tag{7}$$

$$swE^N = E^N \times wE^N \tag{8}$$

Then, user preferences are aggregated into one value, called Aggregated User Score (*AUS*), to be used in the expert system. It is calculated as follows:

$$AUS = swP^N + swC^N + swE^N \tag{9}$$

D. Knowledge base

The knowledge base is a database that stores all the necessary information that the inference engine needs to make decisions. This information may consist of facts or rules that are relevant to the problem domain. For the knowledge base in the proposed system, we need these values:

1) Scene ranking: Managing scene complexity is challenging as many parameters, each with varying values, are attached to every scene. Thus, we introduce the following values: Complexity Value (CV), and Frame Value (FV). In the CV, we assign a weight to each parameter, and then aggregate each mapped weighted value from the mapping scales in Figure 1 for each scene. Thus, CV for Scene i is calculated as follows:

$$CV_i = \frac{(wPoly + wVer + wObj + wLit + wMat)}{5} \tag{10}$$

In this equation, we will use only five parameters from the scene, as the others are not quantitative directly.
To find FV, we need to find the total number of all scene frames as follows:

$$Fram_T = \sum_{i=1}^{n} \mathrm{Fram}_i \tag{11}$$

Then, FV_i is calculated as:

$$FV_i = \frac{Fram_i}{Fram_T} \tag{12}$$

To quantify the scene complexity in one value, we propose an Aggregated Scene Score (ASS) value. ASS for scene i is the sum of the complex value of the scene (CV) and the Frame Value (FV) of that scene:

$$ASS_i = CV_i + FV_i \tag{13}$$

The speed rank of each scene is calculated as:

$$SceneSpeedRank_i = \begin{cases} 1 & \text{if } ASS_i = 0 \\ \lfloor ASS_i \times 10 \rfloor & \text{otherwise} \end{cases} \tag{14}$$

and the total scene speed rank of all scenes, which should be $1 \leq SceneSpeedRank_T \leq 10$, is defined as follows:

$$SceneSpeedRank_T = \frac{\sum_{i=1}^{n} SceneSpeedRank_i}{n} \tag{15}$$

where n is the number of scenes.

2) GPU Ranking: Selecting the best fit GPU to render any scene depends on the GPU availability, user preferences, and scene complexity. As mentioned before, we will consider hypothetical GPUs but with real specifications, as listed in Table I.
To maintain optimal performance, the high-ranked GPU will be assigned tasks that require high computing power, while the low-ranked GPU will be assigned tasks that require relatively less computing power. The rendering tasks will be distributed efficiently across the available GPUs, resulting in faster and more accurate processing.

E. Inference Engine

The inference engine of the system takes the data from the knowledge base, analyzes it, and applies logical rules to it to make decisions accordingly. The rules of the system are explained as follows:

TABLE I
GPUs SPECIFICATIONS AND RANKING.

GPU model	No. of GPUs	No. of cores	Memory	Energy Consumption	Cost	Speed Rank	Cost Rank	Energy Rank
Model 1	10	10400	24GB	350W	1	10	1	1
Model 2	50	8700	20GB	320W	0,9	9	2	2
Model 3	40	8200	16GB	300W	0,8	8	3	3
Model 4	22	7600	16GB	290W	0,7	7	4	4
Model 5	77	6800	12GB	280W	0,6	6	5	5
Model 6	8	6400	11GB	250W	0,5	5	6	6
Model 7	30	5200	8GB	225W	0,4	4	7	7
Model 8	11	4500	6GB	160W	0,3	3	8	8
Model 9	17	1200	4GM	130W	0,2	2	9	9
Model 10	20	870	4GM	110W	0,1	1	10	10

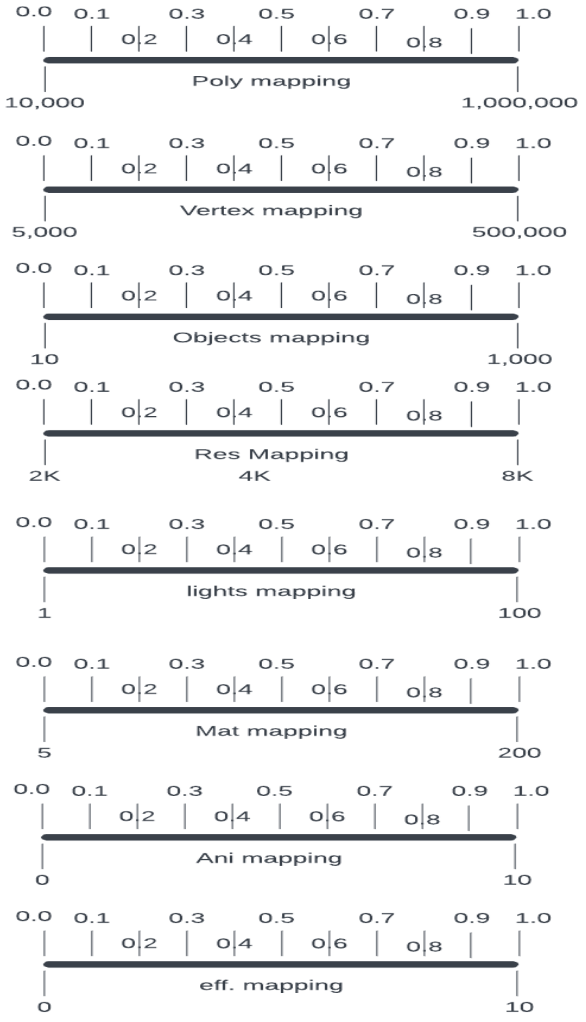

Fig. 1. Normalizing scene parameters into weighted values.

- Rule 1: The total scene speed rank greater than 8
If the total scene speed rank is greater than 8:

- Rank Type: Speed rank
- Rank Value: Maximum of the total scene speed rank and user speed rank.

- Rule 2: AUS less than 5
If AUS is less than 5:

- Rank Type: Speed rank
- Rank Value: The total scene speed rank.

- Rule 3: AUS greater than 5 and the total scene speed rank less than 8

If AUS is greater than 5 and the ultimate rank is less than 8:

- Rank Type: Maximum speed rank, cost rank, and energy rank.
- Rank Value: The value associated with the determined rank type.

- Rule 4: The ultimate value is chosen

If one of the ultimate values is chosen (ultimate speed, ultimate cost saving, or ultimate energy saving):

- Rank Type: The chosen ultimate type
- Rank Value: 10

IV. PERFORMANCE EVALUATION

A. Implementation

The system is coded in Python, using Streamlit, an open-source Python library designed to help create and share custom web apps. The user interface for the proposed system is user-friendly (See Figure 2), where users can send the scenes to be rendered. Mainly it consists of three parts:

- Upload files: Users upload the scenes and GPU specifications files. However, it could be the responsibility of the service provider to upload the specifications of GPUs.
- Select (level of) preferences (performance, cost, and energy).
- Users have the option to give normal preferences. Then, the system will analyze and select the GPU or go with the ultimate option in which the system considers the user's certain desire and ignores the other preferences. In other words, the ultimate option of a specific preference leads to selecting the highest available ranked GPU that corresponds to that preference. However, this option could be disabled by the service provider.

Once a user sends the scene for rendering, the system starts to calculate the AUS and ASS values, specify the required ranks, and accordingly select the GPU to render the scene.

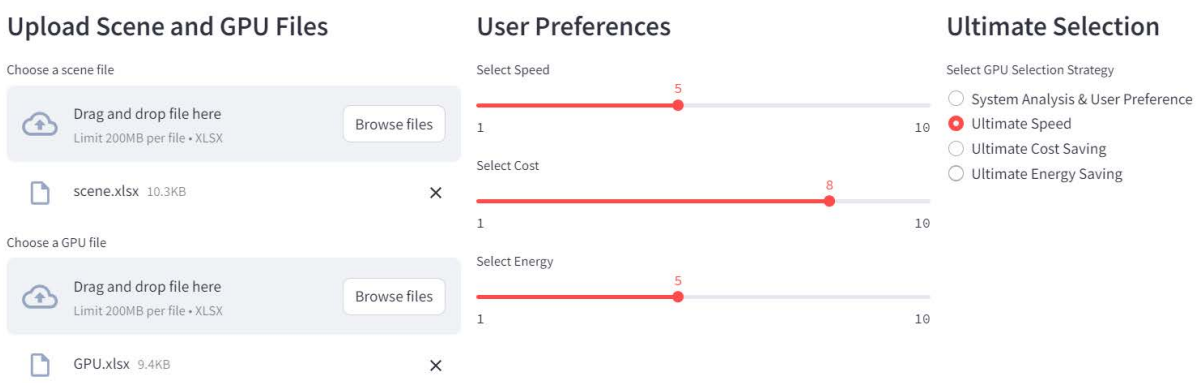

Fig. 2. System Interface.

TABLE II
SYSTEM ANALYSIS FOR DIFFERENT SCENES.

Scene	Poly	Ver	Obj	Lit	Mat	Frames	wPoly	wVer	wObj	wLit	wMat	CV	FV	ASS	Speed Rank
1	715296	394033	765	66	106	56	0,8	0,8	0,8	0,7	0,6	0,74	0,013207547	0,753207547	7
2	413540	36881	939	62	60	171	0,5	0,1	1	0,7	0,3	0,52	0,040330189	0,560330189	5
3	133829	41483	927	24	183	103	0,2	0,1	1	0,3	1	0,52	0,024292453	0,544292453	5
4	789513	442030	604	68	144	345	0,8	0,9	0,6	0,7	0,8	0,76	0,081367925	0,841367925	8
5	210041	240895	675	39	38	218	0,3	0,5	0,7	0,4	0,2	0,42	0,051415094	0,471415094	4
6	854302	164220	176	80	89	335	0,9	0,4	0,2	0,8	0,5	0,56	0,079009434	0,639009434	6
7	914699	69118	168	96	99	81	1	0,2	0,2	1	0,5	0,58	0,019103774	0,599103774	5
8	348537	405149	269	65	182	413	0,4	0,9	0,3	0,7	1	0,66	0,09740566	0,75740566	7
9	971024	190478	576	66	68	193	1	0,4	0,6	0,7	0,4	0,62	0,045518868	0,665518868	6
10	642986	276801	644	82	59	140	0,7	0,6	0,7	0,9	0,3	0,64	0,033018868	0,673018868	6
11	489116	164434	797	1	188	106	0,5	0,4	0,8	0,1	1	0,56	0,025	0,585	5
12	580872	430058	489	79	116	433	0,6	0,9	0,5	0,8	0,6	0,68	0,102122642	0,782122642	7
13	589531	334896	566	90	136	238	0,6	0,7	0,6	0,9	0,7	0,7	0,056132075	0,756132075	7
14	567950	33238	911	98	20	198	0,6	0,1	1	1	0,1	0,56	0,046698113	0,606698113	6
15	767389	384762	376	75	111	223	0,8	0,8	0,4	0,8	0,6	0,68	0,05259434	0,73259434	7
16	220296	342287	653	97	50	54	0,3	0,7	0,7	1	0,3	0,6	0,012735849	0,612735849	6
17	903395	337919	541	73	40	204	1	0,7	0,6	0,8	0,2	0,66	0,048113208	0,708113208	7
18	964116	369861	27	22	10	416	1	0,8	0,1	0,3	0,1	0,46	0,098113208	0,558113208	5
19	859379	148816	980	37	24	240	0,9	0,3	1	0,4	0,1	0,54	0,056603774	0,596603774	5
20	83071	485460	663	49	144	73	0,1	1	0,7	0,5	0,8	0,62	0,017216981	0,637216981	6

B. Results

When running the system, users can try different scenarios and balance their preferences with scene complexity. Different results for the *Rank Type* (Performance, cost, and energy) and *Rank Value* (0 to 10) will be obtained, and the GPU that matches their needs is allocated accordingly.

Figure 2 shows the user interface and how users can specify their preferences, Table II shows some results when rendering scenes with different complexities, and Table III shows that different GPU models are selected based on the users' input.

C. Discussion

In the proposed approach, the results show that different GPUs are selected to do the rendering process for users, even for the same scene. This is because different users have different preferences, and these preferences affect the results.

By analyzing the variables defined in this work, we can come to different points:

- When the User Aggregated score (AUS) is less than 5, it means that the users have no interest in optimizing the performance, cost, or energy. In this case, the system picks the Performance rank with the highest available value.

- When the Aggregated Scenes Score (ASS) is greater than 8, then the system optimizes Performance rank (regarding rendering speed), no matter what the user preferences are, because heavy scenes require significant computational resources.

- When a specific user preference chooses the ultimate option, then the system will pick the rank type of that preference, with the maximum rank value.

However, when the above cases do not apply, then the analysis and GPU matching are dependent on the computation of the *Rank Type* and *Rank Value* of the user preferences. The results look promising in two directions: the proposed approach gives users the tools to cover all their interests, and also offers operators a solution to ensure profitability and sustainability without compromising user preferences.

V. RELATED WORK

This section categorizes the related works into two categories: the current systems and services offered by different operators, and the works that presented cloud-based expert systems.

TABLE III
PREFERENCES TO GPU MATCHING.

Perf	Cost	Energy	Ultimate Perf	Ultimate Cost	Ultimate Energy	Speed Rank	Rank Type	Rank value	GPU
8	4	5	✓	×	×	6	Perf	10	Model 1
7	9	7	×	✓	×	6	Cost	10	Model 10
5	4	6	×	×	✓	6	Energy	10	Model 10
8	4	5	×	×	×	6	Perf	8	Model 3
7	9	7	×	×	×	6	Cost	9	Model 10
5	4	6	×	×	×	6	Energy	6	Model 6
4	5	2	×	×	×	6	Perf	6	Model 5
8	6	6	×	×	×	9	Perf	9	Model 2

A. GPU rendering platforms

Different cloud rendering operators offer GPU services at specific costs. Google offers rendering as a service where the pricing information is offered only for the GPUs as separate resources for defined fees. Extra fees are added for other resources like networking and VM instance pricing. Their services are available in [7]. Microsoft Azure offers a remote rendering service that utilizes Azure's computing power to render complex models in the cloud. The rendered models can then be streamed in real time, allowing users to interact with 3D content seamlessly. Their services are available in [8]. Amazon offers rendering services via their EC2 G3 instances that deliver a powerful combination of CPU, host memory, and GPU capacity. Their rendering services and the cost can be found in [9]. However, although the operators provide rendering services with good performance, they may ignore other factors like energy and overall cost.

B. Cloud-based expert systems

Extensive research has recently been conducted on the Cloud-Based Expert System (CBES) model for decision-making.

In [10], The authors introduced an intelligent system to optimize cloud systems, which models various cloud parameters like VM and SLA configurations to minimize cost and improve overall income. In [11], the authors proposed an expert system for decision making. The system considers parameters from the cloud infrastructure, and the cloud-based applications to make decisions. However, no research explores the use of expert systems in the render farms of cloud data centers. This is an important direction to be addressed in the future to enhance the performance, and reduce the cost and environmental impact of rendering in cloud data centers.

VI. CONCLUSION AND FUTURE WORK

This paper introduces a system for utilizing GPUs in the render farms of cloud data centers. The system aims to minimize cost and energy usage while ensuring consistent and high-quality rendering results. The system enhances cloud data centers' performance by selecting the best-fit resources for rendering. It also has sustainability benefits and positive environmental impacts as it considers energy consumption when dedicating the resources.

The future scope of this work can be summarized in the following directions:

- defining and solving an optimization function for the user preferences with different constraints.
- discussing and testing the serving of many users with limited computation capabilities.
- testing more scene parameters that are not numerically quantified, like the animation difficulty, and showing how such parameters affect the scene-to-GPU matching processes.
- introducing a professional pricing strategy based on different parameters.
- integrating deep learning and fuzzy logic with the system.

ACKNOWLEDGEMENT

This work is partially funded by the Swedish Knowledge Foundation (KKS) under the MARC project, by the Swedish Research Council (VR) under the PSI project, and by the Excellence in Production Research (XPRES), a government funded Strategic Research Area (SRA) within manufacturing engineering in Sweden.

REFERENCES

[1] A. Mehrabi, M. Siekkinen, T. Kämäräinen, and A. yl Jski, "Multi-tier cloudvr: Leveraging edge computing in remote rendered virtual reality," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 17, no. 2, pp. 1–24, 2021.

[2] A. Al-Dulaimy, W. Itani, J. Taheri, and M. Shamseddine, "bwslicer: A bandwidth slicing framework for cloud data centers," *Future Generation Computer Systems*, vol. 112, pp. 767–784, 2020.

[3] A. Inc., "What's the difference between gpus and cpus?." https://aws.amazon.com/compare/the-difference-between-gpus-cpus/, 2024. Accessed: 2024-02-12.

[4] I. N. da Silva and R. A. Flauzino, *Application of Expert Systems: Theoretical and Practical Aspects*. BoD–Books on Demand, 2020.

[5] J. Kastner and S. Hong, "A review of expert systems," *European journal of operational research*, vol. 18, no. 3, pp. 285–292, 1984.

[6] D. Maylawati, W. Darmalaksana, and M. A. Ramdhani, "Systematic design of expert system using unified modelling language," in *IOP Conference Series: Materials Science and Engineering*, vol. 288, p. 012047, IOP Publishing, 2018.

[7] G. LLC., "Gpu pricing." https://cloud.google.com/compute/gpus-pricing, 2024. Accessed: 2024-02-12.

[8] M. Corporation, "Remote rendering pricing." https://azure.microsoft.com/en-us/pricing/details/remote-rendering/, 2024. Accessed: 2024-02-12.

[9] A. Inc., "Amazon ec2 g3 instances: Accelerate your graphics-intensive workloads with powerful gpu instances." https://aws.amazon.com/ec2/instance-types/g3/, 2024. Accessed: 2024-02-12.

[10] A. Bernal, P. Cañizares, A. Núñez, M. Cambronero, and V. Valero, "Iso-cloud: An intelligent system for optimizing the overall income in cloud providers," in *2022 4th International Conference on Computer Communication and the Internet (ICCCI)*, pp. 7–13, IEEE, 2022.

[11] M. Bakator and D. Radosav, "Expert systems in a cloud computing environment model for fast-paced decision making," *JITA-APEIRON*, vol. 13, no. 1, 2017.