Vol. 34, No. 2, 2025

Machine GRAPHICS & VISION

International Journal

Published by
The Institute of Information Technology
Warsaw University of Life Sciences – SGGW
Nowoursynowska 159, 02-776 Warsaw, Poland
in cooperation with
The Association for Image Processing, Poland – TPO

Radar Image Processing Application Based on Space Cloud Computing in Basketball Game Guidance Camera

Jun Song
School of Physical Education, Qilu Normal University, Ji'nan, China
15805310520@163.com

Submitted: 15 Oct 2024 **Accepted:** 27 Mar 2025 **Published:** 19 May 2025

Abstract Capturing and presenting exciting moments is crucial for the audience's experience in basketball game broadcast cameras. However, traditional radar image processing techniques are limited by various factors and cannot meet the demands of modern audiences for high quality, multi angle, and real-time performance. In response to these challenges, an innovative radar image processing system based on space cloud computing has been proposed. Compared with traditional radar image processing systems, the system proposed by the research institute had the best performance, with accuracy, recall, and F1 value reaching 97.08%, 96.88%, and 97.11%, respectively, and a transmission time of only 2.2 seconds; and the stability was greater than 90%, which was about 10% to 25% higher than other systems. In summary, the system proposed by the research institute has brought revolutionary improvements to basketball game guidance and filming through its efficient processing capabilities, accurate image recognition, fast data processing and transmission, and excellent stability. This not only greatly enriches the audience's viewing experience, but also opens up new directions for the development of sports event broadcasting technology. With the further maturity of technology and the continuous expansion of applications, it is expected that this system will play a more important role in future sports event broadcasting, promoting the development of the entire industry towards higher quality and efficiency.

Keywords: competition guide, space cloud computing, SAR, radar image, image recognition.

1. Introduction and state-of-the-art

Basketball games are one of the most popular sports events in the world, and their broadcast quality has a direct impact on the viewing experience of the audience [31]. To ensure high-quality broadcasting, the director's camera often relies on monitoring equipment such as radar to assist in locating and tracking the dynamics of athletes and basketballs when completing the shooting and switching of live footage [17]. This technical support enables the director's camera to perform precise shooting, ensuring real-time capture and smooth transmission of images, thereby enhancing the audience's viewing experience [4,27]. Synthetic Aperture Radar (SAR) is an advanced radar system known for its high resolution, capable of precise remote detection of ground and ocean targets, and real-time imaging [33]. These capabilities have enabled SAR to be widely applied in various fields, such as military reconnaissance, geological surveys, environmental monitoring, etc. Sleem et al. combined the historical time series of SAR with the latest geotechnical data to monitor the causes of ground instability in the Ismailia region. They processed 25 ENVISAT ASAR scenarios and 77 differential interferograms.

The ground instability in Ismailia District was mainly controlled by three basic parameters: groundwater depth, clay layer depth, and expansion potential [30]. To analyze the vegetation status in agricultural monitoring, Kumar derived a new vegetation index from dual polarized SAR data using scattering information from polarization degree and eigenvalue spectrum. The vegetation index of rapeseed had the highest correlation with biophysical variables, with determination coefficients of 0.79, 0.82, and 0.75 for plant area index, vegetation moisture content, and dry biomass, respectively [18]. Hati et al. used SAR data to estimate the aboveground biomass of highly complex mangrove communities on the Indian island of Lothian. The aboveground biomass of Lothian Island ranged from 1.15 tons/hectare to 79.83 tons/hectare, with an average of 20.23 tons/hectare [13]. Kikon et al. used differential SAR interferometry to estimate the frequency of surface deformation in Kohima District, Nagaland State, India in 2018 and 2019, in response to the shortcomings of existing surface deformation detection technologies. The subsidence rate in Koshima County was relatively high in 2018, with higher subsidence rates in the southern and western regions compared to the northern and eastern regions [16].

Although SAR technology is essential in multiple fields with its rapid data processing capabilities, wide monitoring range, and excellent resolution, the large amount of data and high-density data streams it collects pose a huge challenge for real-time data processing [8]. The introduction of cloud computing technology provides a new solution to this problem. Oh et al. proposed a privacy protection data collection method based on mobile cloud computing technology to ensure the secure storage of user information and prevent data from being easily tampered with. The proposed method improved the delay efficiency by 170% [26]. Li conducted research on the Chinese translation of literary works using artificial intelligence and machine learning capabilities on cloud computing platforms to decipher cultural translation. This method revealed the subtle relationship between the Chinese translation of literary works and the content before and after translation by focusing on the semantic relationships between words [20]. Madankan used queue theory to model cloud computing to measure the service quality of computer services. The arrival rate and service rate of the processing server were the two main parameters that affected the performance of the model. By modeling the queue network of the cloud platform, it was easy to determine and measure the service quality of computer services [23]. Ding et al. designed a space-based embedded cloud computing platform to address the limitations of existing cloud computing technologies in fully utilizing storage and computing resources, and to meet the application requirements of space-based intelligent computing and ground-based cloud computing systems. This platform not only performed well in processing large-scale data, but also had significant advantages in ensuring real-time and reliability of computing tasks [9]. Alsharari et al. proposed a cloud computing method based on cloud enterprise resource planning to explore the impact of contingency factors on the public sector in the United Arab Emirates. The calculation results showed that institutional pressure in highly institutionalized environments could

Tab. 1. Summary of relevant studies.

Radar image processing technology	Publication	Research content	Research result
Supervised multi-view hash model	Yan et al. [36]	Enhancing multi-view information by neural networks.	The research methodology significantly outperforms state-of-the-art single-view and multi-view hashing methods.
Group-based nuclear norm and learning graph (GNNLG) model	Yan et al. [38]	In order to collect similarities between patches more efficiently	The proposed method is superior to the current state-of-the-art image denoising methods in terms of both subjective and objective criteria.
Task Adaptive Attention Module for Image Captioning	Yan et al. [37]	To mitigate the misdirection of traditional attention mechanisms	Inserting a task-adaptive attention module into a dummy-transformer-based image caption model improves model performance.
Precision reference-free image quality assessment programme	Yan et al. [41]	To address the shortcomings of the lack of understanding of image distortion in non-referenced image quality assessment (NR IQA)	Aberration classification accuracy of the research method is higher than that of state-of-the-art aberration classification methods.
Multi-feature Fusion and Decomposition (MFD) Framework	Yan et al. [39]	In order to be able to learn more discriminative and robust features of images and to reduce intra-class variation.	Image recognition performance is significantly improved.
Space-based embedded cloud computing platform	Ding et al. [9]	Addressing the shortcomings of existing cloud computing technologies where storage and computing resources cannot be fully utilised	The platform not only excels in processing large-scale image data, but also has significant advantages in ensuring the real-time and reliability of computational tasks.

generate organizational responses, but these responses depended on various aspects of organizational culture and are influenced by it [2]. In order to visually compare the differences between various image processing techniques, the study summarizes them. An overview of the research related to image processing techniques is shown in Table 1.

In summary, the current radar image processing technology generally still faces the

problems of insufficient data processing capacity and limited storage capacity, which can not ensure the real-time capture and smooth transmission of the picture in the basketball game guide camera. In addition, it also has the problems of insufficient resolution and high delay in basketball game broadcasting. Insufficient resolution will lead to blurred images, which cannot clearly show the movement of the athletes and the trajectory of the ball; high latency will affect the real-time performance, and the audience may see images that are inconsistent with the actual event. At the same time, radar signals are susceptible to environmental interference, leading to inaccurate target tracking, and the computational burden of data fusion and processing also increases the complexity of the system. Based on this situation, this study innovatively proposes a system architecture based on space cloud computing technology, and utilizes the flow of state diagrams to implement complex SAR imaging processing flow. Finally, this study proposes a radar image processing system based on space cloud computing architecture and variable scale imaging algorithm. The purpose of this study is to meet the precise shooting needs of basketball games by the director's camera and improve the real-time performance of SAR imaging processing.

This study is divided into three parts. The next part (Section 2) introduces how space cloud computing, SAR imaging algorithms, and the overall system framework are improved, as well as how the radar image processing system is established. The following part (Section 3) describes the performance testing of the new system, and the last part (Section 4) is a conclusion of the article.

2. Methods and materials

In response to the problems existing in the current basketball game broadcast camera, the study first introduces the principle of SAR system radar imaging, and proposes a multi star index based variable scale imaging algorithm by introducing SAR imaging algorithm. In addition, the combination of space cloud computing and multi satellite collaborative work has been utilized, and a SAR overall framework based on space cloud computing has been constructed. Finally, a new radar image processing system based on space cloud computing architecture is established, which is based on the overall framework of SAR using space cloud computing and the variable scale imaging algorithm using multi star indexing.

2.1. Selection and improvement of SAR imaging algorithms

In the field of radar imaging, researchers and technology developers have always pursued high resolution as one of their core goals. High resolution capability can make the observation and recognition of target details clearer, which is particularly important in basketball game guidance cameras [11, 12]. Because clear images not only enhance the visual experience of the audience, but also provide more accurate information on the

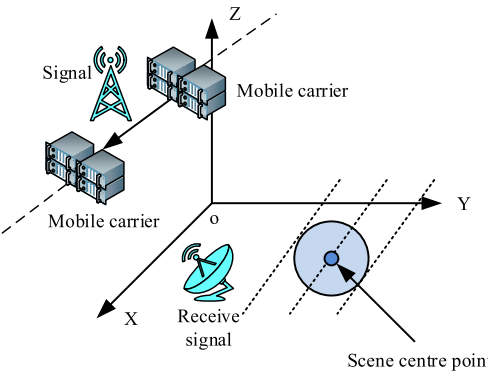


Fig. 1. Radar imaging principles for SAR system.

athlete's position and movement trajectory [44]. SAR systems can significantly improve imaging resolution without increasing the physical antenna size by simulating larger antenna apertures [14]. The radar imaging principle of SAR system is shown in Figure 1. In this Figure, SAR imaging is based on a basic principle: a moving carrier moves along a straight line in a specific direction at a constant speed, continuously emitting pulse signals at a fixed frequency to the target area. At the same time, the receiving device carried by the carrier continuously captures and fully records the echo signals reflected by the target [6]. The SAR imaging algorithm is the core of efficient data processing in SAR systems [3]. In this process, SAR imaging algorithms not only ensure the accuracy and efficiency of data processing, but also play a crucial role in improving the resolution and reliability of images [24]. The widely used SAR imaging algorithm currently is the range Doppler algorithm, however, this algorithm only focuses on the impact of distance curvature on imaging and ignores the influence of distance curvature differences in the scene [21,35]. In response to this situation, this study improved the SAR imaging algorithm and proposed a multi-scale imaging algorithm based on multi star indexing. The data transmission process of this algorithm in the spaceborne radar SAR imaging processing is shown in Figure 2. In this Figure, the algorithm proposed in this study adopts the strategy of distributing data processing tasks from the main satellite to the auxiliary satellite in the imaging processing of spaceborne radar SAR. In the data transmission stage, the main satellite evenly divides the data to be processed into four parts, and then sends the corresponding data segments to the corresponding auxiliary satellites according to their respective serial numbers. The auxiliary satellites involved in this process include satellites numbered 1, 2, 3, and 4, all of which undertake the comprehensive processing of the algorithm in both distance and azimuth directions. In addition, the main satellite is responsible for transmitting the independent variable values

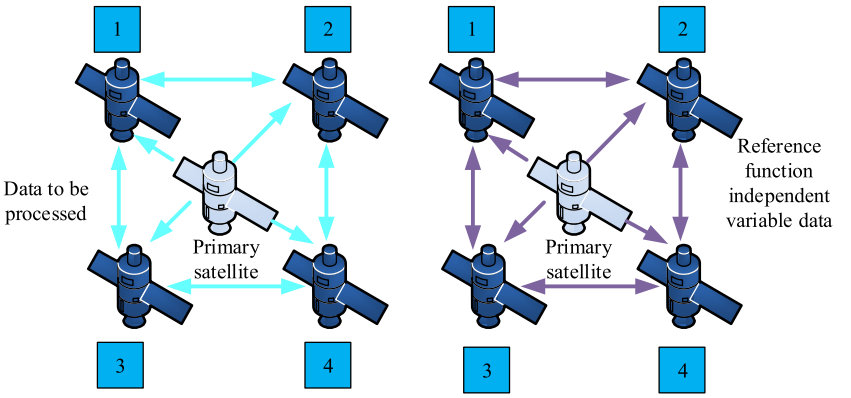


Fig. 2. Data transfer flow during SAR imaging processing.

of all reference functions to these auxiliary satellites separately to ensure consistency and accuracy in data processing.

2.2. Overall framework structure design of SAR based on space cloud computing

Space cloud computing is an emerging technology concept that deploys computing resources in space to achieve real-time processing and analysis of Earth observation data, providing near real-time data processing capabilities. This is particularly important for application scenarios that require rapid response, such as real-time guidance and filming of basketball games [5,7]. The service types of space cloud computing mainly include edge computing as a Service (ECaaS), Infrastructure as a Service (IaaS), Network as a Service (NaaS), Platform as a Service (PaaS) and Software as a Service (SaaS) [1]. These service types can be selected and combined according to the specific needs and application scenarios of users to achieve the best cloud computing solution [43]. However, spaceborne radar SAR imaging technology has extremely high standards for data processing capabilities, which often exceed the capabilities of a single spaceborne processing platform. So it is necessary to achieve excellent data processing capabilities through the collaborative work of multiple satellites [15,42]. Therefore, this study built a multi satellite space cloud computing processing platform based on satellite collaborative work. The structure of the multi satellite space cloud computing processing platform is shown in Figure 3. In this Figure, the multi satellite space cloud computing processing platform proposed by the research institute consists of multiple small satellites, such as the main satellite, small satellite 1, small satellite 2, small satellite 3, and small satellite 4 in Figure 3. These satellites work together to form a network that covers a wider detection area. Satellites transmit data through laser communication technology, which ensures high bandwidth

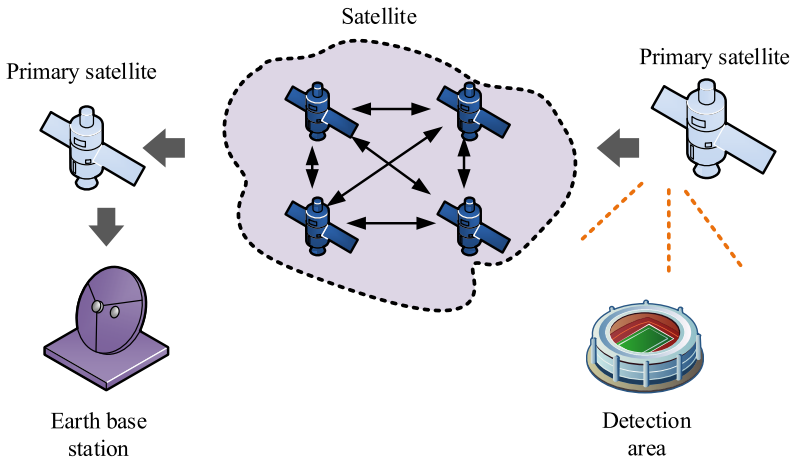


Fig. 3. Architecture of a multi-satellite space cloud computing processing platform.

and low latency. Therefore, satellites can transmit detected data in real-time, ensuring the timeliness of data processing and jointly completing the real-time imaging processing task of the entire spaceborne SAR. Linear Frequency Modulated (LFM) algorithm is a widely used signal processing technique in radar systems. By analyzing the frequency changes of the echo signal, the distance and relative velocity of the target can be determined. It is very suitable for high-resolution imaging processing of spaceborne SAR for long-distance migration [10, 34]. Therefore, based on the multi satellite space cloud computing processing platform, this study introduces the LFM algorithm and utilizes the flow of state diagrams to implement a complex SAR imaging processing flow. The mapping relationship between processes is shown in Figure 4. In this Figure, the processing flow of LFM is implemented through 11 state diagrams. The main satellite and the small satellite cycle three times in three stages, namely the main satellite waiting for reply stage, the small satellite processing subtask stage, and the waiting for the next stage instruction stage. In the first stage, the main satellite is responsible for distributing raw data and task instructions, while the small satellites perform operations such as FFT, linear frequency modulation, and matrix transposition. In the second stage, the main satellite updates tasks according to instructions. In the third stage, the small satellite completes the IFFT and returns the results, while the main satellite collects and confirms the completion of the task, and carries out subsequent operations according to the instructions of the ground control center. Therefore, to meet the task and functional design of master-slave satellites, this study combines a multi satellite space cloud computing processing platform, LFM algorithm, and state diagram flow to construct a SAR overall framework structure based on space cloud computing. The expression for

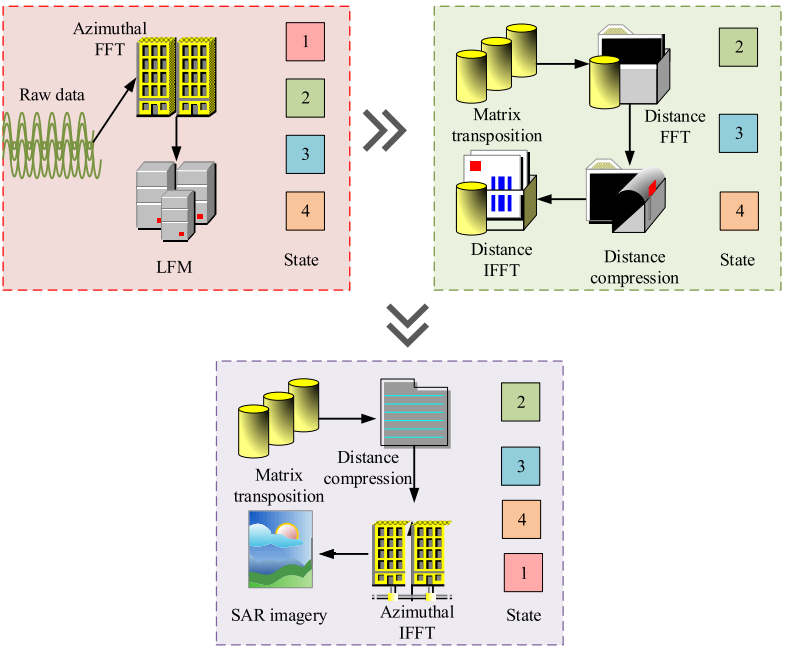


Fig. 4. Mapping between the flow of multi-satellite connectivity state charts and the processing flow of the LFM algorithm.

calculating the linear frequency modulation signal $s(\hat{t}, t_m)$ is shown in equation (1):

$$s(\hat{t}, t_m) = \text{rect}\left(\frac{\hat{t}}{T_p}\right) = \exp\left[j2\pi\left(f_c t + \frac{1}{2}\gamma\hat{t}^2\right)\right], \tag{1}$$

where t, $\hat{t}$ and t_m represent full time, distance fast time, and azimuth slow time, respectively; $\text{rect}(\cdot)$ represents rectangular window function; f_c and γ represent modulation frequency and signal carrier frequency, respectively; T_p represents pulse width. Phase is a fundamental part of a signal and instantaneous phase is used to describe how the frequency of a signal changes over time. The instantaneous phase of the LFM algorithm up converted waveform $\psi(t)$ calculation expression is shown in equation (2):

$$\psi(t) = 2\pi(f_o t + \frac{1}{2}\gamma t^2), \tag{2}$$

where f_o represents the center frequency of the carrier signal, indicates the fundamental frequency of the signal without FM. The expression for calculating the instantaneous

frequency $f(t)$ is shown in equation (3):

$$f(t) = \frac{1}{2\pi}\frac{d}{dt}\psi(t) = f_o + \gamma t\,. \tag{3}$$

The expression for distance resolution N is shown in equation (4):

$$N = \frac{R_m - R_n}{\Delta R}\,, \tag{4}$$

where R_m, R_n, and ΔR represent the maximum distance, minimum distance, and the distance difference between two targets, respectively. The expression for the azimuth target distance $R_a(t_m)$ is shown in equation (5):

$$R_n(t_m) = \sqrt{R_s^2 + (Vt_m - X_n)^2}\,, \tag{5}$$

where V represents the speed of the carrier platform motion; t_m represents slow time sampling; R_s and X_n, respectively, represent the vertical distance of the carrier motion direction and the X coordinate of point n. The expression for the fundamental frequency echo $s_n(t_m)$ is shown in equation (6):

$$s_n(t_m) = \sigma_n e^{-j\frac{4\pi R_n(t_m)}{c} f_c}\,, \tag{6}$$

where σ_n represents the amplitude of the signal. The phase transformation of the echo in the azimuth direction $\varphi_n(t_m)$ is calculated as shown in equation (7):

$$\varphi_n(t_m) = \frac{-4\pi f_c}{c}[R_n(t_m) - R_s]\,, \tag{7}$$

where c represents the speed of light propagation. The expression for calculating the Doppler frequency f_d is shown in equation (8):

$$f_d = \frac{2f_c V}{c}\sin\theta_n\,, \tag{8}$$

where θ_n represents the oblique angle of view. The Doppler bandwidth Δf_d is calculated as shown in equation (9):

$$\Delta f_d = \frac{2V}{D}\,, \tag{9}$$

where D represents the transverse aperture length. The expression for calculating the phase function $H_1(\hat{t}, f_a, R_s)$ by changing the linear frequency modulation scale is shown in equation (10):

$$H_1(\hat{t}, f_a, R_s) = \exp\left[j\pi\gamma_e(f_a, R_B)a(f_a)\left(\hat{t} - \frac{2R(f_a, R_B)}{c}\right)^2\right]\,, \tag{10}$$

where f_a and R_B, respectively, represent the Doppler frequency and the shortest distance from point n to the flight path, and $a(\cdot)$ represents the echo response envelope.

2.3. Construction of radar image processing system based on space cloud computing architecture

When satellites conduct uninterrupted and all-weather monitoring, the obtained satellite image data is very large, and traditional radar image processing systems cannot meet the processing needs of heavy load data [19, 25, 40]. Therefore, based on the powerful computing and analysis capabilities of the SAR overall framework using space cloud computing, as well as the high-resolution advantages of the multi star index based variable scale imaging algorithm, a new radar image processing system based on space cloud computing architecture has been established in this study. The system mainly consists of fiber optic transmission module, DDR3 SDRAM storage module, FFT operation module, multiplication module, and data stream. This design is able to overcome the limitation that the traditional system is unable to handle the huge satellite image data. Through the cooperation of fiber optic transmission, DDR3 SDRAM storage, FFT operation and multiplication module, it can not only effectively improve the speed and accuracy of data processing, optimize the management of data flow, but also solve the problem of high load data processing and transmission. These improvements enable the new system to have higher processing efficiency, image quality and real-time performance, which can show obvious advantages when facing large data volume and complex tasks.

The structure of the DDR3 SDRAM storage module is shown in Figure 5, in which this storage module in the new research system mainly consists of four front-end First In First Out (FIFO) data structures, four back-end FIFOs, one interface FIFO, and four partition memories. The write operation of data in partition memory 1 is only performed by the front-end FIFO1 and transmitted through the interface FIFO, ultimately stored in the corresponding partition memory 1 address of DDR3 SDRAM. Similarly, data in partitioned memory 1 can only be read through backend FIFO1. The data writing and reading process for partition memory 2, partition memory 3, and partition memory 4 follows the same operational flow as that for partition 1.

The working state flow of the DDR3 SDRAM storage module during data read and write operations is shown in Figure 6. The DDR3 SDRAM storage module first enters an idle state before performing data read and write operations, and while refreshing parameters, checks whether the front-end FIFO is waiting to process data. If there is data in the frontend FIFO, the module will check if there is enough space in the current area to store this data. If there is enough space, the module will read data from the front-end FIFO and perform DDR3 write operations. After writing is completed, the module will update the parameters. If there is no data in the front-end FIFO or the current area space is insufficient, the module will check whether there is data in the back-end FIFO. If there is, the module will detect whether there is data inside DDR3 and perform a read operation. After the read operation is completed, the module updates its parameters and prepares to switch to the next area.

The algorithm processing flow in the radar image processing system based on space

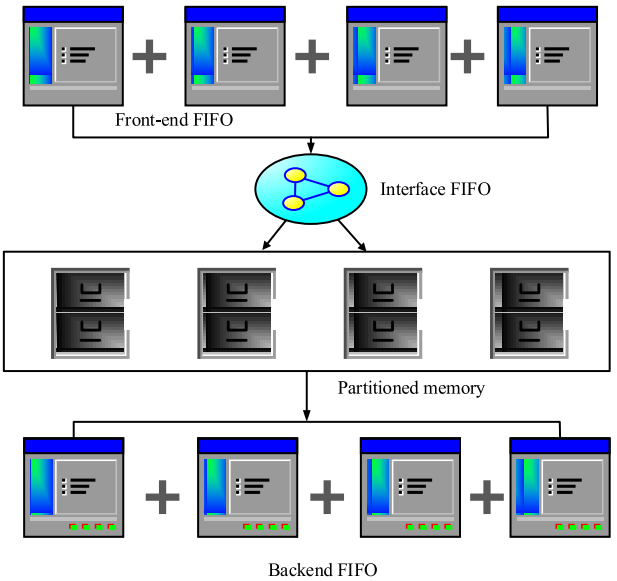


Fig. 5. Architecture of DDR3 SDRAM memory modules.

cloud computing architecture is shown in Figure 7. The new radar image processing system constructed by the research institute performs algorithm processing in five stages. Firstly, the upper computer software sends SAR data and related variables to the first processing unit via optical fiber. Next, the system performs azimuth FFT and performs complex multiplication using the reference function H1. Then, the data is transposed, subjected to distance compression and correction, and multiplied by H2. Afterwards, the data is converted to the time domain through IFFT and stored in DDR3 SDRAM. Finally, the imaging data is transmitted back to the upper computer through optical fibers for image display and subsequent processing. The definition of FFT is shown in equation (11):

$$X(k) = \sum_{n=0}^{N-1} x(n) \exp(-jnk2\pi/N)\,, \tag{11}$$

where N is the conversion length. The inverse definition of FFT $x(n)$ is calculated as shown in equation (12).

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) \exp\left(j2\pi \frac{nk}{N}\right)\,. \tag{12}$$

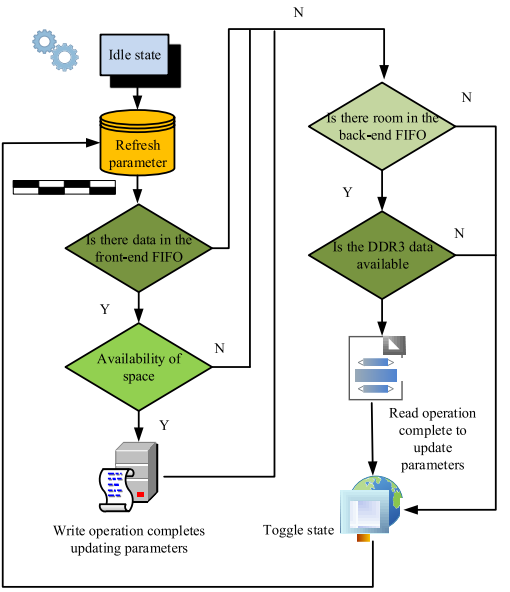


Fig. 6. DDR3 SDRAM memory module operating state flow.

Radar image processing systems based on space cloud computing architectures usually also face high time complexity and space complexity. In terms of time complexity, SAR image processing requires a large number of computations, such as Fourier transform, etc., and the computation time grows significantly as the data volume increases. In order to improve efficiency, the system utilises distributed computing resources to process large data volumes in parallel and reduce the computational burden of a single node. In terms of space complexity, the storage requirement of SAR images is proportional to the resolution, and the huge storage volume of high-resolution images requires efficient storage systems (e.g., DDR3 SDRAM) and data compression techniques to manage the data. It is recommended to use at least 8-core processors with at least 3.0 GHz clocks, such as AMD Ryzen 9 7950X or Intel Core i9-13900K, paired with at least 64 GB of RAM (DDR4/DDR5). For GPU, NVIDIA A100 Tensor Core or NVIDIA V100 is recommended with 40 GB/80 GB HBM2 memory and 6912 CUDA cores to support parallel computing and deep learning acceleration to improve image processing efficiency. For storage, use high-performance SSDs such as Samsung 970 PRO 2 TB or Intel Optane SSD 905P 1.5 TB with DDR4 3200 MHz or DDR5 4800 MHz memory (64 GB or 128 GB) to ensure fast data access and high-bandwidth through Fibre Channel or InfiniBand data transfer.

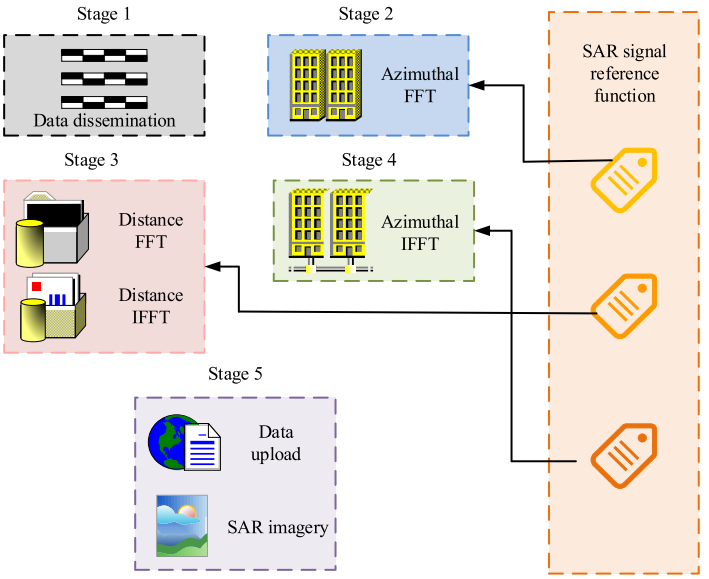


Fig. 7. Algorithmic process handling in radar image processing system based on space cloud computing architecture.

3. Results

To verify the performance of the new SAR imaging algorithm and the radar image processing system based on space cloud computing architecture, this study first sets up a suitable experimental environment and preprocessed the test data. Secondly, the proposed method was tested through comparative testing of the same type and multi criteria performance testing. After completion, the new research system was simulated and tested using actual basketball game guidance camera data.

3.1. SAR imaging algorithm performance testing

This study was based on the Windows operating system, and the XC7VX485T chip from the Virtex-7 FPGA series of Xilinx was selected as the core processor. The CPU was Intel Core i9-13900K and the GPU was NVIDIA V100. The test data came from the GMTSAR dataset [28, 29] and a synthetic DORiSAR dataset generated via SAR imaging simulator, using parameters similar to Sentinel-1 acquisition geometry [22]. These datasets were divided into training and testing sets in a 6 : 4 ratio. Matlab [32] was used to set 5 point targets, where pulses were emitted and echo signals were generated.

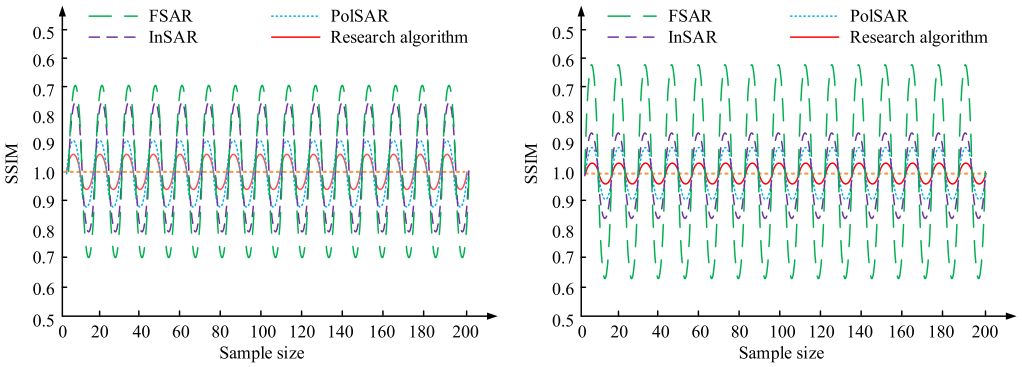
Tab. 2. Important parameter settings during the imaging process.

Serial number	Parameter name	Numerical value
1	Radar carrier frequencies	5.4×10^9
2	Azimuthal sampling rate	1257.02
3	Radar sampling rate	3.26×10^7
4	Pulse time width	4.22×10^{-5}
5	Distance-directed frequency	7.19×10^{11}
6	Nearest radar to target	9.91×10^5
7	Radar equivalent velocity	7064
8	Doppler centre frequency	−6893
9	Data window start time	6.61×10^{-3}

Then, the echo signals were filtered and denoised, and the images were generated by frequency domain transformation using FFT. A variable scale imaging algorithm based on multiple star indexes was applied to improve the image resolution and clarity. The data was accessed through a DDR3 SDRAM memory module and a fibre optic transmission module ensureed fast data flow. Finally, the training set optimised the algorithm and the test set verified the performance to ensure the stability and real-time response of the system under different conditions.

The important parameter settings during the imaging process are shown in Table 2. Based on these values, this study introduced popular SAR imaging algorithms of the same type for comparison, namely the Focused Synthetic Aperture Radar (FSAR), the Synthetic Aperture Radar Interferometer SAR (InSAR), and the Polarimetric Synthetic Aperture Radar (PolSAR).

Firstly, the image quality evaluation index, Structural Similarity Index (SSIM), was used to compare different imaging algorithms. The test results are shown in Figure 8. In this Figure it can be seen that, whether in the GMTSAR dataset or the DORiSAR dataset, the SAR imaging algorithm performed the best, followed by the PolSAR algorithm and InSAR algorithm, and the FSAR algorithm performed the worst. In Figure 8a, the maximum SSIM values for FSAR algorithm, InSAR algorithm, PolSAR algorithm, and the proposed algorithm in the GMTSAR dataset were 0.69, 0.75, 0.89, and 0.96, respectively. In Figure 8b, the maximum SSIM values for the four algorithms in the DORiSAR dataset were 0.61, 0.88, 0.91, and 0.99, respectively. The above data indicates that the research algorithm had significant advantages in maintaining image structure and quality. It improved the SAR imaging algorithm by using a multi star index variable scale imaging method, which could effectively enhance the algorithm's efficiency in maintaining image quality. In addition, SAR images with high SSIM values



a GMTSAR **b** DORiSAR

Fig. 8. Comparison of SSIM performance tests of different SAR imaging algorithms.

provided reliable data support for subsequent experiments, ensuring the accuracy and reliability of the experimental results.

To confirm the resolution capability of the proposed algorithm, the four SAR imaging algorithms were tested using image clarity as the indicator. The test results are shown in Figure 9. In the images shown in this Figure it can be seen that the result from the research algorithm (Fig. 9b) was visually very close to that of the original image (Fig. 9a), and it was almost impossible to observe significant quality differences with the naked eye. The clarity of radar images obtained by PolSAR algorithm, InSAR algorithm, and FSAR algorithm (Figs. 9c-9e) differed significantly from the original images, especially the radar image obtained by FSAR algorithm (Fig. 9e). From this it follows that the research algorithm performed better in radar image resolution than PolSAR, InSAR, and FSAR algorithms, demonstrating its potential and advantages in the field of radar image processing.

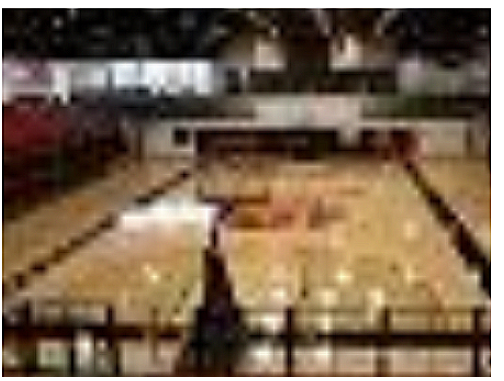
In addition, to explore the real-time capture capability of the proposed algorithm, the data processing speed of the four algorithms mentioned above was also tested. The test results are shown in Figure 10. It can be seen in this Figure that the research algorithm exhibited fast data processing speed in both datasets. Within the range of increasing the sample size from 20 to 200, the data processing speed of all algorithms decreased, but the research algorithm still maintained a relatively fast speed. In Figure 10a, the shortest data processing speeds for FSAR algorithm, InSAR algorithm, PolSAR algorithm, and the research algorithm were 14.9 s, 14.2 s, 12.7 s, and 5.6 s, respectively. In Figure 10b, the shortest data processing speeds for FSAR algorithm, InSAR algorithm, PolSAR algorithm, and the research algorithm were 14.5 s, 14.0 s, 10.9 s, and 5.0 s, respectively. The above data indicate that the research algorithm had good efficiency, scalability, and excellent real-time performance in processing large-scale SAR data.

a Original

b Research algorithm

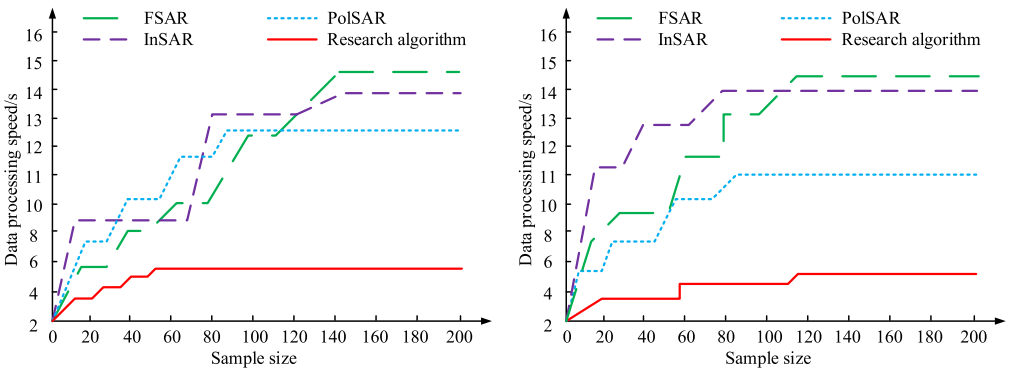
c PolSAR

d InSAR



e FSAR

Fig. 9. Radar image clarity experiment comparison results.



a GMTSAR **b** DORiSAR

Fig. 10. Comparison results of data processing speed test for different SAR algorithms.

3.2. Simulation testing of radar image processing system based on space cloud computing architecture

This study used actual observation data obtained from spaceborne SAR for simulation analysis. The SAR image data had 8192 pixels in both distance and azimuth directions, and each pixel was composed of a real and imaginary part represented by a 32-bit single precision floating-point number, forming complex data. Therefore, the total capacity of the entire dataset reached 512MB, and the dataset was divided into training and testing sets in a 6 : 4 ratio. To better reflect the imaging performance of the research system, this study also introduced traditional Extended Azimuth Non Linear Variable Scaling Radar Processing System (EANLVSRS), Doppler Radar Processing System (DRPS), and Frequency Modulated Continuous Wave Radar (FMCW) image processing systems. The ground target is selected with a 3×3 uniform point array. The simulation scene is shown in Figure 11.

Based on the simulation scenario shown in Figure 11 and the actual observation data obtained, the data was normalized and cleaned. Random sampling was conducted for multi index testing with precision, recall, F1 value, and transmission time as reference indicators. The test results are shown in Table 3. According to these results, the research system demonstrated high accuracy, high recall, and high F1 score on both the training and testing sets, while also having the shortest transmission time. On the training set, its accuracy, recall, and F1 score were 96.84%, 95.59%, and 96.64%, respectively, with a transmission time of 2.6 seconds; On the test set, its accuracy, recall, and F1 score were 97.08%, 96.88%, and 97.11%, respectively, with a transmission time of 2.2 seconds. The EANLVSRS system performed the worst on both the training and testing sets. On the training set, its accuracy, recall, and F1 score were only 68.36%, 67.08%, and 68.55%,

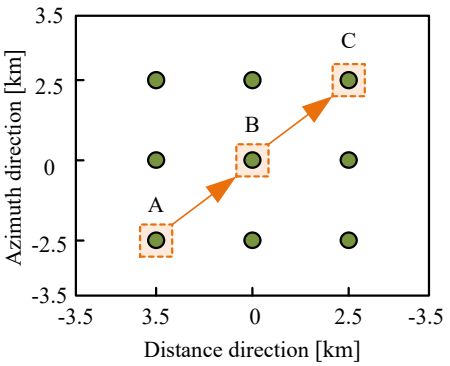


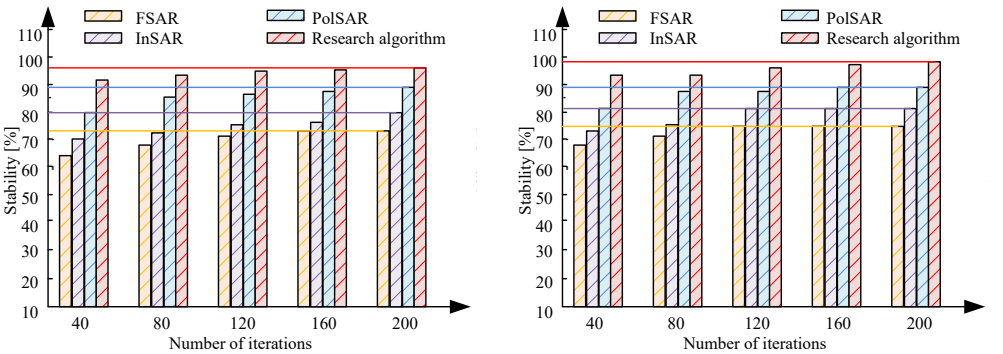
Fig. 11. Simulation scenario.

Tab. 3. Comparison of test results with various indicators. A: accuracy, R: recall, F1: F1 score.

Data set	System	A [%]	R [%]	F1 [%]	Transmission time [s]
Training set	EANLVSRS	68.36	67.08	68.55	15.2
	DRPS	72.57	71.55	72.85	12.1
	FMCW	82.45	80.22	83.53	11.6
	Research system	96.84	95.59	96.64	2.6
Test set	EANLVSRS	69.65	68.37	70.11	14.6
	DRPS	73.54	72.08	73.02	11.9
	FMCW	83.55	82.53	82.88	11.3
	Research system	97.08	96.88	97.11	2.2

respectively, with a transmission time of 15.2 seconds; On the test set, its accuracy, recall, and F1 score were 69.65%, 68.37%, and 70.11%, respectively, with a transmission time of 14.6 seconds. Compared with the research system, the accuracy, recall, and F1 value of the EANLVSRS system had been reduced by about 28%, and the transmission time had been extended by about 13 seconds. From this, the real-time data processing capability of space cloud computing and the ability to simplify complex processes through multi satellite connection state diagrams effectively improved the responsiveness of radar image processing systems, promoting the research system to have good multi index performance effects.

To confirm the stability of the system during operation, we have continued to test the performance of the four systems mentioned above. The test results are shown in Figure 12. As it can be seen in the graphs shown in this Figure, the stability of all four



a GMTSAR **b** DORiSAR

Fig. 12. Comparison of results for stability of different radar image systems.

Tab. 4. Comparison of running time of different systems.

Data set	Systems	1st [s]	2nd [s]	3rd [s]	4th [s]	5th [s]	6th [s]	7th [s]
Training set	EANLVSRS	15.2	15.6	15.8	16.1	16.2	16.8	17.0
	DRPS	12.1	12.2	12.5	12.7	12.8	13.0	13.5
	FMCW	11.6	11.8	12.1	12.2	12.5	12.6	12.8
	Research system	2.6	2.8	3.0	3.2	3.4	3.6	3.7
Test set	EANLVSRS	14.6	14.8	14.9	15.3	15.8	15.6	16.0
	DRPS	11.9	12.1	12.3	12.8	13.0	13.2	13.4
	FMCW	11.3	11.5	11.6	11.7	11.9	12.1	12.3
	Research system	2.2	2.3	2.4	2.6	2.7	2.9	3.0

systems slowly increased with the increase of iteration times. The research system had the best stability effect, with stability greater than 90% in both the training and testing sets. The highest stability of the EANLVSRS system, DRPS system, and FMCW system was only 74.9%, 81.3%, and 89.6%, respectively. Compared with the research system, the stability of the other three systems decreased by about 10% to 25%. From the above data it follows that the research system not only had high accuracy in processing radar images, but also had significantly better stability than other systems, indicating that it had higher reliability and robustness in practical applications.

In this study we have also compared the operational complexity of the different systems mentioned above. The test results are shown in Table 4. The running times of different systems gradually increased with the number of executions. The running time of EANLVSRS, DRPS, FMCW, and research system in the training set increased by

Tab. 5. Metrics test results for different models. Resource consumption rate (Resource CR) refers to the relative proportion of computing resources (such as CPU, memory, electrical energy, etc.) consumed by the system during the execution of image processing tasks, reflecting the efficiency and overhead of the system operation.

Environments	Systems	MSE	MAE	Resource CR [%]	Accuracy [%]
Rainy	EANLVSRS	0.078	0.081	23.33	80.83
	DRPS	0.072	0.066	20.16	84.04
	FMCW	0.065	0.057	18.65	91.24
	Research system	0.033	0.028	7.43	96.89
Cloudy	EANLVSRS	0.063	0.064	20.44	84.12
	DRPS	0.057	0.052	19.32	87.84
	FMCW	0.041	0.045	17.25	91.58
	Research system	0.028	0.029	6.84	97.71
Sunny	EANLVSRS	0.054	0.052	20.13	86.49
	DRPS	0.048	0.049	18.87	89.36
	FMCW	0.038	0.040	15.36	92.75
	Research system	0.021	0.022	6.65	98.23

1.8 s, 1.4 s, 1.2 s, and 1.1 s, respectively; The running time in the test set increased by 1.4 s, 1.5 s, 1.0 s, and 0.8 s, respectively. However, the increase was very small and did not affect the overall performance of the system. The research system had the shortest running time, only 3.0 seconds; FMCW system was second, with a duration of 12.3 s; The EANLVSRS system had the longest running time, which was 16.0 s. From this it follows that the research system was computationally simple and could efficiently process basketball game guidance camera data. The main reason for this was the powerful computing and analysis capabilities of the SAR overall framework based on space cloud computing, which had greatly improved the transmission and operation efficiency.

Finally, in order to investigate the effect of environment on the system, in this study we have also compared the above four systems for rainy, cloudy and sunny environments using Mean Squared Error (MSE), Mean Absolute Error (MAE), resource consumption rate and image processing accuracy as reference indicators. The test results are shown in Table 5. As can be seen in this Table, whether in rainy, cloudy or sunny environments, the EANLVSRS system has the worst performance of the four systems in terms of various metrics. The performance of DRPS, FMCW and the radar image processing system with the new space cloud computing architecture proposed by the Institute varies from low to high, and intuitively, it can be found that the research system has the lowest MSE value of 0.021, the lowest MAE value of 0.022, the lowest resource consumption rate is 6.65% and the highest image processing accuracy is 98.23%. Also, the rainy, cloudy and

sunny environments have less impact on the system performance. In summary, the new system proposed in the study has the best overall performance, and has more stable and superior recognition capability compared with the same type of radar image processing system. The wide applicability and robustness of the research system are verified.

4. Conclusion

With the continuous advancement of technology, the technology of directing and filming basketball games is also constantly developing. To meet the modern audience's demand for high-definition and real-time competition images, a new radar image processing system based on space cloud computing architecture is established by studying the powerful computing capabilities of the SAR overall framework based on space cloud computing and the high-resolution advantages of the multi star index based variable scale imaging algorithm. Whether in the GMTSAR dataset or the DORiSAR dataset, the research algorithm performed the best, with a maximum SSIM value of 0.99, which was very close to 1. The generated radar images were visually very close to the original images, and there was almost no obvious quality difference visible to the naked eye, indicating its significant advantage in maintaining image structure and quality. The simulation test results showed that the real-time data processing capability of space cloud computing and the ability to simplify complex processes through multi satellite connection state diagrams effectively enhance the responsiveness of radar image processing systems. The transmission time of the research system was only 2.2 seconds. As the number of runs increased, the increase in transmission time was very small, which did not affect the overall performance of the system. The complexity of operation was low, the calculation was simple, and it could efficiently process basketball game guidance and camera data. In summary, the excellent performance of the research system in terms of stability and real-time performance could make it a breakthrough technology in the field of basketball game guidance and filming. The research system's high-resolution imaging capability, combined with the efficient data processing of space cloud computing, provides a new technological platform for sports event broadcasting. However, the study did not include more complex basketball game director camera data for testing. Subsequent research can further optimize the algorithm by reducing the system's dependence on computing resources and incorporating more complex basketball game guidance camera data to enhance the comprehensiveness of the study. There are also a number of challenges in the actual deployment and maintenance of the research system. First, the deployment cost is high, mainly due to the need for high-performance GPUs, storage devices, and cloud computing infrastructure, which require large initial investments in equipment and technology. Maintenance challenges mainly include hardware management (e.g., GPU cooling, storage scaling), data security and system optimisation, especially the need to optimise algorithms to reduce dependence on computing resources

when faced with complex data. In terms of commercial potential, the system is capable of delivering high-definition, real-time game footage, which meets the high technological requirements of sports events and therefore has wide market demand. As the technology matures and cost and maintenance issues are resolved, the system is expected to become one of the core technologies for event broadcasting.

Author's declarations

Conflict of interest

The author has no conflict of interest to report.

Data availability

The information on the data sources are included in the manuscript in Section 3.1.

References

[1] B. Alouffi, M. Hasnain, A. Alharbi, W. Alosaimi, H. Alyami, et al. A systematic literature review on cloud computing security: Threats and mitigation strategies. *IEEE Access* 9(5):57792–57807, 2021. doi:10.1109/ACCESS.2021.3073203.

[2] N. M. Alsharari. Cloud computing and ERP assimilation in the public sector: institutional perspectives. *Transforming Government: People, Process and Policy* 16(1):97–109, 2022. doi:10.1108/TG-04-2021-0069.

[3] R. M. Asiyabi, A. Ghorbanian, S. N. Tameh, M. Amani, S. Jin, et al. Synthetic aperture radar (SAR) for ocean: A review. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 16(4):9106–9138, 2023. doi:10.1109/JSTARS.2023.3310363.

[4] A. R. A. Besari, A. A. Saputra, W. H. Chin, Kurnianingsih, and N. Kubota. Finger joint angle estimation with visual attention for rehabilitation support: A case study of the chopsticks manipulation test. *IEEE Access* 10(94):91316–91331, 2022. doi:10.1109/ACCESS.2022.3201894.

[5] K. Cao, S. Hu, Y. Shi, A. W. Colombo, S. Karnouskos, et al. A survey on edge and edge-cloud computing assisted cyber-physical systems. *IEEE Transactions on Industrial Informatics* 17(11):7806–7819, 2021. doi:10.1109/TII.2021.3073066.

[6] J. Chen, M. Xing, H. Yu, B. Liang, J. Peng, et al. Motion compensation/autofocus in airborne Synthetic Aperture Radar: A review. *IEEE Geoscience and Remote Sensing Magazine* 10(1):185–206, 2022. doi:10.1109/MGRS.2021.3113982.

[7] M. Dai, Z. Su, R. Li, and S. Yu. A software-defined-networking-enabled approach for edge-cloud computing in the Internet of Things. *IEEE Network* 35(5):66–73, 2021. doi:10.1109/MNET.101.2100052.

[8] Y. Deng. Optimising enterprise financial sharing process using cloud computing and big data approaches. *International Journal of Grid and Utility Computing* 13(4):272–281, 2022. doi:10.1504/ijguc.2022.10049066.

[9] Y. Ding, Y. Shao, and J. Zhang. A cloud computing platform technology based on space-based intelligent information network. *Journal of Physics: Conference Series* 2203(1):12–22, 2022. doi:10.1088/1742-6596/2203/1/012042.

[10] N. Fu, S. Yun, B. Han, and L. Qiao. Sub-Nyquist measurement of LFM pulse stream based on signal separation and parameter matching. *IEEE Transactions on Instrumentation and Measurement* 72(6506015):1–15, 2023. doi:10.1109/TIM.2023.3328704.

[11] X. Gao, S. Roy, and G. Xing. MIMO-SAR: A hierarchical high-resolution imaging algorithm for mmWave FMCW radar in autonomous driving. *IEEE Transactions on Vehicular Technology* 70(8):7322–7334, 2021. doi:10.1109/TVT.2021.3092355.

[12] M. Gottinger. Coherent automotive radar networks: The next generation of radar-based imaging and mapping. *IEEE Journal of Microwaves* 1(1):149–163, 2021. doi:10.1109/JMW.2020.3034475.

[13] J. P. Hati, A. Mukhopadhyay, and N. R. Chaube. Estimation of above ground biomass with synthetic aperture radar (SAR) data in Lothian Island, Sundarbans, India. *Journal of the Indian Society of Remote Sensing* 52(4):757–769, 2024. doi:10.1007/s12524-023-01788-9.

[14] Y. Huang, Z. Chen, C. Wen, J. Li, X. G. Xia, et al. An efficient radio frequency interference mitigation algorithm in real Synthetic Aperture Radar data. *IEEE Transactions on Geoscience and Remote Sensing* 60(4):5224912, 2022. doi:10.1109/TGRS.2022.3155068.

[15] M. Jing and G. Zhang. Location of synthetic aperture radar imagery via range history. *IEEE Transactions on Geoscience and Remote Sensing* 60:5232112, 2022. doi:10.1109/TGRS.2022.3192984.

[16] N. Kikon, D. Kumar, and S. A. Ahmed. Spaceborne synthetic aperture radar (SAR) earth observations for surface deformation analysis and mapping. *Arabian Journal of Geosciences* 15:1131, 2022. doi:10.1007/s12517-022-10370-5.

[17] A. Kittel, P. Larkin, and N. Elsworthy. Transfer of 360 virtual reality and match broadcast video-based tests to on-field decision-making. *Science and Medicine in Football* 5(1):79–86, 2021. doi:10.1080/24733938.2020.1802506.

[18] D. Kumar. Urban objects detection from C-band synthetic aperture radar (SAR) satellite images through simulating filter properties. *Scientific Reports* 11(1):21–30, 2021. doi:10.1038/s41598-021-85121-9.

[19] B. Li and X. Xu. Application of Artificial Intelligence in basketball sport. *Journal of Education, Health and Sport* 11(7):54–67, 2021. doi:10.12775/JEHS.2021.11.07.005.

[20] K. Li. Ding Xilin's Chinese translation of The Man of Destiny: Reading Shaw through IBM Artificial Intelligence and Cloud Computing Platform. *Shaw* 41(1):151–166, 2021. doi:10.5325/SHAW.41.1.0151.

[21] N. Li, Z. Lv, and Z. Guo. Pulse RFI mitigation in Synthetic Aperture Radar data via a three-step approach: Location, Notch, and Recovery. *IEEE Transactions on Geoscience and Remote Sensing* 60:5225617, 2022. doi:10.1109/TGRS.2022.3161368.

[22] A. Liu, N. Wang, D. Dettmering, Z. Li, M. Schmidt, et al. Using DORIS data for validating real-time GNSS ionosphere maps. *Advances in Space Research* 71(1):115–128, 2023. doi:10.1016/j.asr.2023.01.050.

[23] A. Madankan. The response time analysis of queuing model in cloud computing environment. *Computer Science Journal of Moldova* 30(4):64–76, 2022. doi:10.56415/csjm.v30.04.

[24] D. Mao. Forward-looking geometric configuration optimization design for spaceborne-airborne multistatic Synthetic Aperture Radar. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 14(4):8033–8047, 2021. doi:10.1109/JSTARS.2021.3103802.

[25] A. Nuriddinov. Use of digital sports technologies in sports television. *American Journal Of Social Sciences And Humanity Research* 3(11):208–219, 2023. doi:10.37547/ajsshr/Volume03Issue11-22. https://inlibrary.uz/index.php/ajsshr/article/view/38358.

[26] E. N. Oh, M. R. Baharon, S. M. W. M. S. M. M. Yassin, A. Idris, and A. MacDermott. Preserving data privacy in mobile cloud computing using enhanced homomorphic encryption scheme. *Journal of Physics: Conference Series* 2319:012024, 2022. doi:10.1088/1742-6596/2319/1/012024.

[27] P. Pięta, H. Jegierski, P. Babiuch, M. Jegierski, M. Płaza, et al. Automated classification of virtual reality user motions using a motion atlas and machine learning approach. *IEEE Access* 12:94584–94609, 2024. doi:10.1109/ACCESS.2024.3424930.

[28] D. Sandwell, R. Mellors, X. Tong, M. Wei, and P. Wessel. Open radar interferometry software for mapping surface deformation. *Eos, Transactions American Geophysical Union* 92(28):234–234, 2011. doi:10.1029/2011EO280002.

[29] D. Sandwell, X. E. Xu, R. Mellors, X. Tong, and M. M. Wei. An InSAR processing system based on GMT. GMTSAR, 2010. https://topex.ucsd.edu/gmtsar/.

[30] T. A. Sleem, A. M. Hegazi, and A. M. Zaki. Integration of historical synthetic aperture radar (SAR) and geotechnical data for monitoring ground instability: a case study of Ismailia District, Egypt. *Arabian Journal of Geosciences* 15(1):100, 2022. doi:10.1007/s12517-021-09142-4.

[31] E. F. Smirnov, N. S. Ivanov, M. Zavgorodnii, I. Y. Khramov, and V. V. Chernyshova. Recognition of the basketball players position using live cameras. In: *2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus)*, pp. 695–698, 2021. doi:10.1109/ElConRus51938.2021.9396214.

[32] The MathWorks, Inc. MATLAB. Natick, MA, USA. https://www.mathworks.com.

[33] N. Tomii. Infrastructure monitoring by satellite based synthetic aperture radar (SAR) in JAXA. *Artificial Intelligence and Data Science* 4(1):3–8, 2023. doi:10.11532/jsceiii.4.L1_3.

[34] Z. Wei, N. Fu, S. Jiang, X. Li, and L. Qiao. Parameter measurement of LFM signal with FRI sampling and nuclear norm denoising. *IEEE Transactions on Instrumentation and Measurement* 71:2002417, 2022. doi:10.1109/TIM.2022.3158986.

[35] G. Xu, B. Zhang, H. Yu, J. Chen, M. Xing, et al. Sparse Synthetic Aperture Radar imaging from compressed sensing and machine learning: Theories, applications, and trends. *IEEE Geoscience and Remote Sensing Magazine* 10(4):32–69, 2022. doi:10.1109/MGRS.2022.3218801.

[36] C. Yan, B. Gong, Y. Wei, and Y. Gao. Deep multi-view enhancement hashing for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43(4):1445–1451, 2020. doi:10.1109/TPAMI.2020.2975798.

[37] C. Yan, Y. Hao, L. Li, J. Yin, A. Liu, et al. Task-adaptive attention for image captioning. *IEEE Transactions on Circuits and Systems for Video Technology* 32(1):43–51, 2021. doi:10.1109/TCSVT.2021.3067449.

[38] C. Yan, Z. Li, Y. Zhang, Y. Liu, X. Ji, et al. Depth image denoising using nuclear norm and learning graph model. *ACM Transactions on Multimedia Computing Communications and Applications* 16(4):1–17, 2020. doi:https://doi.org/10.1145/3404374.

[39] C. Yan, L. Meng, L. Li, J. Zhang, J. Yin, et al. Age-invariant face recognition by multi-feature fusion and decomposition with self-attention. *ACM Transactions on Multimedia Computing Communications and Applications* 18(1):1–18, 2021. doi:https://doi.org/10.1145/3472810.

[40] C. Yan, Y. Sun, H. Zhong, C. Zhu, Z. Zhu, et al. Review of omnimedia content quality evaluation. *Journal of Signal Processing* 38(6):1111–1143, 2022. doi:10.16798/j.issn.1003-0530.2022.06.001. https://signal.ejournal.org.cn/article/doi/10.16798/j.issn.1003-0530.2022.06.001.

[41] C. Yan, T. Teng, Y. Liu, Y. Zhang, H. Wang, et al. Precise no-reference image quality evaluation based on distortion identification. *ACM Transactions on Multimedia Computing Communications and Applications* 17(3s):110, 2021. doi:10.1145/3468872.

[42] M. Younis, F. Q. de Almeida, M. Villano, S. Huber, G. Krieger, et al. Digital beamforming for spaceborne Reflector-Based Synthetic Aperture Radar, Part 1: Basic imaging modes. *IEEE Geoscience and Remote Sensing Magazine* 9(3):8–25, 2021. doi:10.1109/MGRS.2021.3060543.

[43] Y. Yuan, S. Chen, Y. Zhang, S. Wang, and S. Zhang. Nonlinear chirp scaling imaging method for three-dimensional foresight linear array maneuvering SAR. *IEEE Transactions on Geoscience and Remote Sensing* 60:5233913, 2022. doi:10.1109/TGRS.2022.3200973.

[44] Z. Zhang, L. Li, G. Cong, H. Yin, Y. Gao, et al. From speaker to dubber: Movie dubbing with prosody and duration consistency learning. In: *Proceedings of the 32nd ACM International Conference on Multimedia*, MM '24, p. 7523–7532. Association for Computing Machinery, New York, NY, USA, 2024. doi:10.1145/3664647.3680777.

Simple Derivation of the Hermite Bicubic Patch using Tensor Product

Vaclav Skala[1,2]
[1]Department of Computer Science and Engineering, University of West Bohemia, Pilsen, Czech Republic
[2]Department of Military Robotics, University of Defence, Brno, Czech Republic
www.VaclavSkala.eu

Submitted: 23 Apr 2025 **Accepted:** 2 Jun 2025 **Published:** 14 Jun 2025

Abstract Bicubic parametric patches are widely used in various geometric applications. These patches are critical in CAD/CAM systems, which are applied in the automotive industry and mechanical and civil engineering. Commonly, Hermite, Bézier, Coons, or NURBS patches are employed in practice. However, the construction of the Hermite bicubic patch is often not easy to explain formally. This contribution presents a new formal method for constructing the Hermite bicubic plate based on the tensor product approach.

Keywords: tensor product, Hermite curve, Hermite patch, interpolation, parametric patches, Kronecker product.

1. Introduction

This contribution introduces a simplified approach to deriving the Hermite bicubic parametric patch. A clear and formal derivation is essential for a proper understanding, particularly in computer graphics and geometric modeling courses, where only the final mathematical form is typically shown. The standard derivation of the Hermite form is often considered complex.

The method presented here is based on the tensor product with linear operators. It is straightforward, easy to follow, and well-suited for introductory courses. The Bézier parametric patch $S(u, v)$, as defined by Bézier [2], is based on the tensor product of Bézier curves:

$$S(u, v) = C(u) \otimes C(v)\,.$$

In general, cubic parametric curves and bicubic patches are discussed in works such as Cogen [3], Goldman [5], Prautzsch [9], Holliday [6], and Rockwood [10].

It should be noted that while the boundary curves of a bicubic patch are cubic, the diagonal and anti-diagonal curves (for $u = v$ and $u = 1 - v$) are of degree 6. If the degree is limited to 3, additional conditions must be applied. Such constraints were formulated for Hermite patches in Skala [14] and Bézier patches in Kolcun [7] and Skala [13]. A geometric interpretation of the diagonal in a Bézier volume was explored by Holliday and Farin [6]. Triangular patches are discussed in Farin [4].

2. Tensor product

The tensor product [17] is not commonly used in basic courses, yet it is a powerful and versatile tool. It defines a non-commutative product of two vectors, $\mathbf{v} = [v_1, v_2, \ldots, v_n]^T$ and $\mathbf{w} = [w_1, w_2, \ldots, w_m]^T$, as:

$$\mathbf{v} \otimes \mathbf{w} = \begin{bmatrix} v_1 w_1 & v_1 w_2 & \cdots & v_1 w_m \\ v_2 w_1 & v_2 w_2 & \cdots & v_2 w_m \\ \vdots & \vdots & \ddots & \vdots \\ v_n w_1 & v_n w_2 & \ldots & v_n w_m \end{bmatrix} \tag{1}$$

The Kronecker product [15], named after Leopold Kronecker, generalizes the outer product and is a specific case of the tensor product. The Kronecker product of two matrices $\mathbf{A}$ and $\mathbf{B}$ is defined as:

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{bmatrix} \otimes \begin{bmatrix} b_{1,1} & b_{1,2} \\ b_{2,1} & b_{2,2} \end{bmatrix} =$$

$$\begin{bmatrix} a_{1,1} \begin{bmatrix} b_{1,1} & b_{1,2} \\ b_{2,1} & b_{2,2} \end{bmatrix} & a_{1,2} \begin{bmatrix} b_{1,1} & b_{1,2} \\ b_{2,1} & b_{2,2} \end{bmatrix} \\ a_{2,1} \begin{bmatrix} b_{1,1} & b_{1,2} \\ b_{2,1} & b_{2,2} \end{bmatrix} & a_{2,2} \begin{bmatrix} b_{1,1} & b_{1,2} \\ b_{2,1} & b_{2,2} \end{bmatrix} \end{bmatrix} = \begin{bmatrix} a_{1,1}b_{1,1} & a_{1,1}b_{1,2} & a_{1,2}b_{1,1} & a_{1,2}b_{1,2} \\ a_{1,1}b_{2,1} & a_{1,1}b_{2,2} & a_{1,2}b_{2,1} & a_{1,2}b_{2,2} \\ a_{2,1}b_{1,1} & a_{2,1}b_{1,2} & a_{2,2}b_{1,1} & a_{2,2}b_{1,2} \\ a_{2,1}b_{2,1} & a_{2,1}b_{2,2} & a_{2,2}b_{2,1} & a_{2,2}b_{2,2} \end{bmatrix} . \tag{2}$$

Applying the tensor product to differential operators yields [12]:

$$\begin{bmatrix} 1 \\ \frac{\partial}{\partial u} \end{bmatrix} \otimes \begin{bmatrix} 1 \\ \frac{\partial}{\partial v} \end{bmatrix} = \begin{bmatrix} 1 & \frac{\partial}{\partial v} \\ \frac{\partial}{\partial u} & \frac{\partial}{\partial u}(\frac{\partial}{\partial v}) \end{bmatrix} = \begin{bmatrix} 1 & \frac{\partial}{\partial v} \\ \frac{\partial}{\partial u} & \frac{\partial^2}{\partial u \partial v} \end{bmatrix} . \tag{3}$$

Both the tensor and the Kronecker products are multilinear [16] and can also be applied to functions [8].

3. Hermite curve using tensor product

The Hermite parametric cubic curve segment uses two end-points x_1, x_2 and two tangential vectors x_3, x_4 of the cubic segment end-points, see Fig. 1.

The position of the point $x(u)$ is given by (4):

$$x(u) = a_1 u^3 + a_2 u^2 + a_3 u + a_4 \,, \quad x(u) = \sum_{i=1}^{4} a_i u^{4-i} \,, \quad u \in < 0, 1 > , \tag{4}$$

and the tangent vector $x'(u) = \frac{dx(u)}{du}$ is given by (5):

$$x^{(u)}(u) = 3a_1 u^2 + 2a_2 u + a_3 \,, \quad x^{(u)}(u) = \sum_{i=1}^{3} (4-i)\ a_i u^{3-i} \,, \quad u \in < 0, 1 > \,. \tag{5}$$

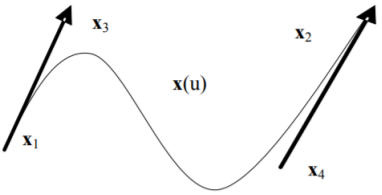


Fig. 1. Hermite cubic curve (tangential vectors shortened); source: own.

The Eq. (4) can be rewritten using the *dot product* as:

$$x(u) = [a_1, a_2, a_3, a_4]^T \; [u^3, u^2, u, 1] = \mathbf{a}^T \mathbf{u} \,. \tag{6}$$

Solving (4) and (5) for the curve segment end-points, i.e. $u = 0$ and $u = 1$, the following system of linear equations is obtained:

$$x(0) = a_4 \,, \quad x(1) = a_1 + a_2 + a_3 + a_4 \,, \quad x^{(u)}(0) = a_3 \,, \quad x^{(u)}(1) = 3a_1 + 2a_2 + a_3 \,, \tag{7}$$

where $x^{(u)} = \frac{\partial x}{\partial u}$.

This leads to a system of equations for the unknown coefficients $\mathbf{a} = [a_1, a_2, a_3, a_4]^T$ for the given end-points property $\boldsymbol{\xi} = [x(0), x(1), x^{(u)}(0), x^{(u)}(1)]^T \stackrel{\text{def}}{=} [x_1, x_2, x_1^{(u)}, x_2^{(u)}]^T$.

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_1^{(u)} \\ x_2^{(u)} \end{bmatrix} \,, \quad \mathbf{B}\mathbf{a} = \boldsymbol{\xi} \,. \tag{8}$$

Solving the linear system of equations $\mathbf{B}\mathbf{a} = \boldsymbol{\xi}$, Eq. (7), the coefficients of the Hermite form are obtained. Then

$$x(u) = a_1 u^3 + a_2 u^2 + a_3 u + a_4 = (\mathbf{B}^{-1}\boldsymbol{\xi})^T \mathbf{u} = \boldsymbol{\xi}^T \mathbf{B}^{-T} \mathbf{u} \,, \tag{9}$$

where $\mathbf{u} = [u^3, u^2, u, 1]^T$, $\mathbf{B}^{-T}$ is the transposed inverse matrix.

Now, the Hermite parametric curve is then described as:

$$x(u) = \boldsymbol{\xi}^T \mathbf{M}_H \mathbf{u} = \boldsymbol{\xi}^T \begin{bmatrix} 2 & -3 & 0 & 1 \\ -2 & 3 & 0 & 0 \\ 1 & -2 & 1 & 0 \\ 1 & -1 & 0 & 0 \end{bmatrix} \mathbf{u} = \mathbf{u}^T \mathbf{M}_H^T \boldsymbol{\xi} \,, \tag{10}$$

where $\mathbf{u} = [u^3, u^2, u, 1]^T$, $\mathbf{M}_H = \mathbf{B}^{-1}$ is the matrix of the Hermite form, and $\boldsymbol{\xi} = [x(0), x(1), x^{(u)}(0), x^{(u)}(1)]^T \equiv [x_1, x_2, x_1^{(u)}, x_2^{(u)}]^T$ are the control values of the curve $x(u)$.

It should be noted that the Equation (10) represents only the $x(u)$-coordinate and for the other coordinates, i.e. $y(u)$, $z(u)$, it is similar.

Generally, for the E^3 case, for a curve $\mathbf{C}(u) = [x(u), y(u), z(u)]^T$ we can write:

$$\mathbf{C}(u) = [\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3, \mathbf{P}_4]^T \; \mathbf{M}_H \; [u^3, u^2, u, 1] = [\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3, \mathbf{P}_4]^T \; \mathbf{M}_H \; \mathbf{u}\,, \tag{11}$$

where $\mathbf{P}_1 = [x_1, y_1, z_1]^T$, $\mathbf{P}_2 = [x_2, y_2, z_2]^T$ are vectors of the curve end-points, $\mathbf{P}_3 = [x_1^{(u)}, y_1^{(u)}, z_1^{(u)}]^T$, $\mathbf{P}_4 = [x_2^{(u)}, y_2^{(u)}, z_2^{(u)}]^T$ are vectors of the tangential vectors at the curve end-points and $\mathbf{u} = [u^3, u^2, u, 1]^T$.

The Equation (10) can be rewritten as:

$$\mathbf{C}(u) = [\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3, \mathbf{P}_4]^T \; \mathbf{M}_H \; \begin{bmatrix} u^3 \\ u^2 \\ u \\ 1 \end{bmatrix}\,, \tag{12}$$

i.e.

$$\begin{bmatrix} x(u) \\ y(u) \\ z(u) \end{bmatrix} = \begin{bmatrix} x_1^{(u)} & x_2^{(u)} & x_3^{(u)} & x_4^{(u)} \\ y_1^{(u)} & y_2^{(u)} & y_3^{(u)} & y_4^{(u)} \\ z_1^{(u)} & z_2^{(u)} & z_3^{(u)} & z_4^{(u)} \end{bmatrix} \; \mathbf{M}_H \; \begin{bmatrix} u^3 \\ u^2 \\ u \\ 1 \end{bmatrix}\,. \tag{13}$$

Note that the Eq. (10) is formally valid also for the Bézier, Catmul, Ferguson, etc. curves; however, the control vector $\boldsymbol{\xi}$ has different properties.

The Bézier curve of the degree n is defined as:

$${}^{(B)}x(u) = \sum_{i=0}^{n} x_i \binom{n}{i} u^i (1-u)^{n-i}\,, \tag{14}$$

and the tangential vectors are defined at the end-points as:

$$x^{(u)}(0) = n(x_1 - x_0) \quad x^{(u)}(1) = n(x_n - x_{n-1})\,. \tag{15}$$

There is a direct connection between the Hermite and Bézier forms. Therefore, the Hermite, Bézier, Ferguson, etc., curves are mutually convertible, see Anand [1].

It should be noted that an invertible matrix $\mathbf{M}_{H\to B}$ (4×4) exists, which transforms the Hermite form to the Bézier form:

$${}^{(B)}x(u) = \mathbf{M}_{H\to B} \; {}^{(H)}x(u)\,, \tag{16}$$

where ${}^{(B)}x(u)$, resp. ${}^{(H)}x(u)$, means the x-coordinate of the Hermite cubic curve, resp. Bézier cubic curve, see Anand [1]. Continuity conditions were also studied in Skala [11].

4. Hermite patch using tensor product

The two-dimensional case of the multi-variate Hermite interpolation is the Hermite bicubic patch. The bicubic parametric patch for the x-coordinate is defined as:

$$x(u,v) = \left(\sum_{i=1}^{4} a_i u^{4-i}\right)\left(\sum_{j=1}^{4} b_j v^{4-j}\right) = \sum_{i=1}^{4}\sum_{j=1}^{4} a_i b_j \; u^{4-i} v^{4-j}\,, \tag{17}$$

$$x(u,v) = \mathbf{u}^T \mathbf{S}\; \mathbf{v}\,, \quad s_{i,j} = a_i b_j\,, \tag{18}$$

where: $\mathbf{u} = [u^3, u^2, u, 1]^T$, $\mathbf{v} = [v^3, v^2, v, 1]^T$ and the matrix $\mathbf{S}$ (4×4) has the $s_{i,j}$ elements. Similarly, for the $y(u,v)$ and $z(u,v)$ coordinates.

Using the tensor product on functions, a simple formula is obtained [12]:

$$x(u,v) = x(u) \otimes x(v)\,. \tag{19}$$

The Eq. (18) describes a parametric patch $x(u,v)$. Each point of the curve $x(u)$ in (10) is parameterized by the second parameter v as:

$$x(u,v) = \mathbf{u}^T \mathbf{M}_H^T \boldsymbol{\xi}(\mathbf{v})\,, \tag{20}$$

where: $\boldsymbol{\xi}(\mathbf{v}) = [x_1(v), x_2(v), x_1^{(v)}(u), x_2^{(v)}(u)]^T$. It should be noted that all elements of the vector $\boldsymbol{\xi}(\mathbf{v})$ are the Hermite curves again. It means, that

$$\begin{aligned} x_1(v) &= [x_{11}, x_{12}, x_{11}^{(v)}, x_{12}^{(v)}]\mathbf{M}_H\mathbf{v}\,,\\ x_2(v) &= [x_{21}, x_{22}, x_{21}^{(v)}, x_{22}^{(v)}]\mathbf{M}_H\mathbf{v}\,,\\ x_1^{(v)}(u) &= [x_{11}^{(u)}, x_{12}^{(u)}, x_{11}^{(uv)}, x_{12}^{(uv)}]\mathbf{M}_H\mathbf{v}\,,\\ x_2^{(v)}(u) &= [x_{21}^{(u)}, x_{22}^{(u)}, x_{21}^{(uv)}, x_{22}^{(uv)}]\mathbf{M}_H\mathbf{v}\,, \end{aligned} \tag{21}$$

where $x^{(uv)} \stackrel{\text{def}}{=} \frac{\partial^2 x}{\partial u \partial v}$ and $\mathbf{v} = [v^3, v^2, v, 1]^T$.

Now, the Hermite patch $x(u,v)$ for the x-coordinate can be rewritten as:

$$x(u,v) = \mathbf{u}^T \mathbf{M}_H^T \begin{bmatrix} x_{11} & x_{12} & x_{11}^{(v)} & x_{12}^{(v)} \\ x_{21} & x_{22} & x_{21}^{(v)} & x_{22}^{(v)} \\ x_{11}^{(u)} & x_{12}^{(u)} & x_{11}^{(uv)} & x_{12}^{(uv)} \\ x_{21}^{(u)} & x_{22}^{(u)} & x_{21}^{(uv)} & x_{22}^{(uv)} \end{bmatrix} \mathbf{M}_H\mathbf{v}\,, \tag{22}$$

or using more compact form:

$$x(u,v) = \mathbf{u}^T \mathbf{M}_H^T\; \mathbf{X}\; \mathbf{M}_H\mathbf{v}\,, \tag{23}$$

where the matrix $\mathbf{X}$ is the matrix of the control values of the Hermite patch form. It can be seen, that it is the biquadratic form. This formal notation is common for the other bicubic patches, e.g., for the Bézier, Ferguson, etc.

5. Hermite bicubic plate using tensor product

The Hermite bicubic plate can be expressed by using tensor product as:

$$\mathbf{S}(u,v) = \mathbf{C}(u) \otimes \mathbf{C}(v)\,. \tag{24}$$

Using the tensor product and more compact form with the block matrix notation:

$$x(u,v) = \mathbf{u}^T \mathbf{M}_H^T \begin{bmatrix} x_{11} & x_{12} & x_{11}^{(v)} & x_{12}^{(v)} \\ x_{21} & x_{22} & x_{21}^{(v)} & x_{22}^{(v)} \\ x_{11}^{(u)} & x_{12}^{(u)} & x_{11}^{(uv)} & x_{12}^{(uv)} \\ x_{21}^{(u)} & x_{22}^{(u)} & x_{21}^{(uv)} & x_{22}^{(uv)} \end{bmatrix} \mathbf{M}_H \mathbf{v}\,. \tag{25}$$

The Eq. (25) can be rewritten to:

$$x(u,v) = \mathbf{u}^T \mathbf{M}_H^T \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \\ x_{41} & x_{42} & x_{43} & x_{44} \end{bmatrix} \mathbf{M}_H \mathbf{v}\,, \tag{26}$$

where x_{ij} are the control values, see Fig. 2.

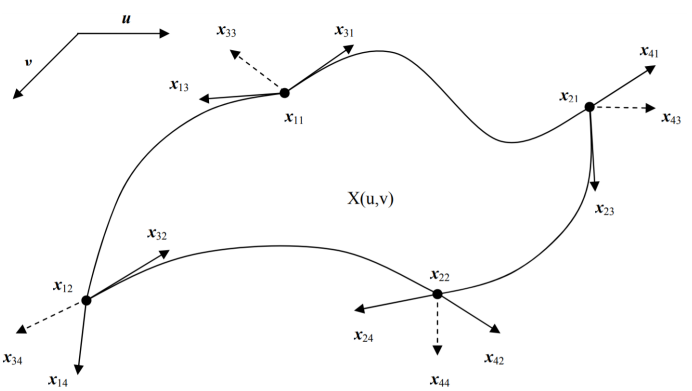


Fig. 2. Hermite bi-cubic patch (tangential and twist vectors scaled); source: own.

Now, using the tensor product and a submatrix 4×4 of the patch end-points a more "compact form" describing the Hermite patch is obtained as:

$$\mathbf{P}(u,v) = \mathbf{u}^T \mathbf{M}_H^T \begin{bmatrix} 1 & \frac{\partial}{\partial v} \\ \frac{\partial}{\partial u} & \frac{\partial^2}{\partial u \partial v} \end{bmatrix} \otimes \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} \mathbf{M}_H \mathbf{v} =$$

$$\mathbf{u}^T \mathbf{M}_H^T \begin{bmatrix} \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} & \frac{\partial}{\partial v} \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} \\ \frac{\partial}{\partial u} \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} & \frac{\partial^2}{\partial u \partial v} \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} \end{bmatrix} \mathbf{M}_H \mathbf{v} \,. \tag{27}$$

If the differential tensor operator (3) is applied on the Hermit bicubic corners, the matrix form is obtained as follows:

$$\begin{bmatrix} 1 & \frac{\partial}{\partial v} \\ \frac{\partial}{\partial u} & \frac{\partial^2}{\partial u \partial v} \end{bmatrix} \otimes \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} & \frac{\partial}{\partial v} \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} \\ \frac{\partial}{\partial u} \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} & \frac{\partial^2}{\partial u \partial v} \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} \end{bmatrix} =$$

$$\begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} & \frac{\partial}{\partial u}\mathbf{P}_{11} & \frac{\partial}{\partial u}\mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} & \frac{\partial}{\partial u}\mathbf{P}_{21} & \frac{\partial}{\partial u}\mathbf{P}_{22} \\ \frac{\partial}{\partial v}\mathbf{P}_{11} & \frac{\partial}{\partial v}\mathbf{P}_{12} & \frac{\partial^2}{\partial u \partial v}\mathbf{P}_{11} & \frac{\partial^2}{\partial u \partial v}\mathbf{P}_{12} \\ \frac{\partial}{\partial v}\mathbf{P}_{21} & \frac{\partial}{\partial v}\mathbf{P}_{22} & \frac{\partial^2}{\partial u \partial v}\mathbf{P}_{21} & \frac{\partial^2}{\partial u \partial v}\mathbf{P}_{22} \end{bmatrix} . \tag{28}$$

It can be seen that this matrix clearly shows the Hermite form properties.

It should be noted that the Hermite bicubic parametric patch can be converted to the Bézier bicubic patch similarly to the case of cubic curves, see Anand [1].

Using a more formally general form:

$$\begin{aligned} \mathbf{S}(u,v) &= \left(\mathbf{u}^T \mathbf{M}_H^T \begin{bmatrix} 1 \\ \frac{\partial}{\partial u} \end{bmatrix} \right) \otimes \left(\begin{bmatrix} 1 & \frac{\partial}{\partial v} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} \right) \mathbf{M}_H \mathbf{v} \\ &= \mathbf{u}^T \mathbf{M}_H^T \left(\begin{bmatrix} 1 & \frac{\partial}{\partial v} \\ \frac{\partial}{\partial u} & \frac{\partial^2}{\partial u \partial v} \end{bmatrix} \otimes \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} \right) \mathbf{M}_H \mathbf{v} \,. \end{aligned} \tag{29}$$

Then using algebraic manipulations, the final formula is obtained:

$$\mathbf{S}(u,v) = \mathbf{u}^T \mathbf{M}_H^T \begin{bmatrix} \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} & \frac{\partial}{\partial v} \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} \\ \frac{\partial}{\partial u} \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} & \frac{\partial^2}{\partial u \partial v} \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} \end{bmatrix} \mathbf{M}_H \mathbf{v} \,. \tag{30}$$

It should be noted that Eq. (30) can be rewritten to a more general form valid for the Bézier, Catmull-Rom, etc. patches, as:

$$x(u,v) = \mathbf{u}^T\mathbf{M}_F^T\ \mathbf{X}\ \mathbf{M}_F\mathbf{v}\,,\ y(u,v) = \mathbf{u}^T\mathbf{M}_F^T\ \mathbf{Y}\ \mathbf{M}_F\mathbf{v}\,,\ z(u,v) = \mathbf{u}^T\mathbf{M}_F^T\ \mathbf{Z}\ \mathbf{M}_F\mathbf{v}\,, \quad (31)$$

where $\mathbf{M}_F$ is a matrix of the form, i.e. Hermite, Bézier, Catmull-Rom, etc., and $\mathbf{X}$, $\mathbf{Y}$, and $\mathbf{Z}$ are the matrices of the control values of the form used.

In the Hermite patch case, parameterization of the x-coordinate is given as:

$$x(u,v) = \mathbf{u}^T\mathbf{M}_H^T \begin{bmatrix} x_{11} & x_{12} & x_{11}^{(v)} & x_{12}^{(v)} \\ x_{21} & x_{22} & x_{21}^{(v)} & x_{22}^{(v)} \\ x_{11}^{(u)} & x_{12}^{(u)} & x_{11}^{(uv)} & x_{12}^{(uv)} \\ x_{21}^{(u)} & x_{22}^{(u)} & x_{21}^{(uv)} & x_{22}^{(uv)} \end{bmatrix} \mathbf{M}_H\mathbf{v}\,; \quad (32)$$

then (32) can be formally simplified further:

$$x(u,v) = \mathbf{u}^T\mathbf{M}_H^T \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \\ x_{41} & x_{42} & x_{43} & x_{44} \end{bmatrix} \mathbf{M}_H\mathbf{v}\,, \quad (33)$$

where x_{ij} represent the control values, as shown in Fig. 2, and similarly for $y(u,v)$ and $z(u,v)$ coordinates.

This matrix clearly illustrates the properties of the Hermite form.

6. Conclusion

This contribution presents an alternative to the Hermite cubic curve and Hermite bicubic patch derivation using the tensor product. The tensor product can be applied not only to vectors and matrices but also to functions.

In many computer graphics courses, only the vector or matrix form is shown, without a detailed derivation of the formulas or a detailed, "boring and lengthy" derivation.

Using the tensor matrix operations makes the derivation of the Hermite form clearer, especially for the Hermite bicubic patch.

Acknowledgement

The author acknowledges the valuable suggestions provided by colleagues and students at the University of West Bohemia in Pilsen and the University of Defence in Brno. Appreciation is also extended to Erasmus students enrolled in courses. Special thanks are due to the anonymous reviewers for their critical comments and constructive recommendations, which significantly improved the quality of this paper.

The work presented in this paper has not been supported by external funding.

References

[1] V. Anand. *Computer Graphics and Geometric Modeling for Engineers*. 1st edn. John Wiley & Sons, Inc., USA, 1993.

[2] P. Bézier. *The Mathematical Basis of the UNIURF CAD System*. Butterworth-Heinemann, 1986. doi:10.1016/C2013-0-01005-5.

[3] E. Cohen, R. F. Riesenfeld, and G. Elber. *Geometric Modeling with Splines: An Introduction*. A K Peters/CRC Press, 2019. doi:10.1201/9781439864203.

[4] G. Farin. Bézier triangles. In: G. Farin (Ed.), *Curves and Surfaces for Computer-Aided Geometric Design*, 3rd edn., chap. 18, pp. 321–351. Academic Press, Boston, 1993. doi:10.1016/B978-0-12-249052-1.50023-4.

[5] R. Goldman. *An Integrated Introduction to Computer Graphics and Geometric Modeling*. 1st edn. CRC Press, Inc., USA, 2009. doi:10.1201/9781439803356.

[6] D. J. Holliday and G. E. Farin. Geometric interpretation of the diagonal of a tensor-product Bézier volume. *Computer Aided Geometric Design* 16(8):837–840, 1999. doi:10.1016/S0167-8396(99)00004-7.

[7] A. Kolcun. Biquadratic S-Patch in Bézier form. In: *WSCG 2011 Communication Papers Proceedings – Proc. 19th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*, vol. 19, pp. 201–207, 2011. http://wscg.zcu.cz/WSCG2011/!_2011_WSCG-Short_Papers.pdf.

[8] N. Mochizuki. The tensor product of function algebras. *Tohoku Mathematical Journal* 17(2):139–146, 1965. doi:10.2748/tmj/1178243579.

[9] H. Prautzsch and W. Boehm. *Geometric Concepts for Geometric Design*. A K Peters/CRC Press, 1993. doi:10.1201/9781315275475.

[10] A. Rockwood and P. Chambers. *Interactive Curves and Surfaces: A Multimedia Tutorial on CAGD*. 1st edn. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1996.

[11] V. Skala. New geometric continuity solution of parametric surfaces. *AIP Conference Proceedings* 1558:2500–2503, 2013. doi:10.1063/1.4826048.

[12] V. Skala. Hermite parametric bicubic patch defined by the tensor product. In: *Computational Science and Its Applications – ICCSA 2022*, vol. 13376 of *Lecture Notes in Computer Science*, pp. 228–235. Springer, 2022. doi:10.1007/978-3-031-10450-3_18.

[13] V. Skala and V. Ondracka. BS-Patch: Constrained Bézier parametric patch. *WSEAS Transactions on Mathematics* 12(5):598–607, 2013. https://wseas.com/journals/articles.php?id=5799.

[14] V. Skala, M. Smolik, and L. Karlicek. HS-Patch: A new Hermite smart bicubic patch modification. *International Journal of Mathematics and Computers in Simulation* 8:292–299, 2014. http://www.naun.org/main/NAUN/mcs/2014/a282002-086.pdf.

[15] Wikipedia contributors. Kronecker product. Wikipedia, The Free Encyclopedia, 2021. https://en.wikipedia.org/wiki/Kronecker_product. [Accessed: 7 Oct 2021].

[16] Wikipedia contributors. Multilinear polynomial. Wikipedia, The Free Encyclopedia, 2021. https://en.wikipedia.org/wiki/Multilinear_polynomial. [Accessed: 2 Oct 2021].

[17] Wikipedia contributors. Tensor product. Wikipedia, The Free Encyclopedia, 2021. https://en.wikipedia.org/wiki/Tensor_product. [Accessed: 7 Oct 2021].

Vaclav Skala is currently a professor at the Dept. of Computer Science and Engineering, University of West Bohemia (UWB), Pilsen [Plzen] and Dept. of Military Robotics, Faculty of Military Technology, University of Defence, Brno. He has been with Brunel University in London, U.K., Gavle University, Sweden, Moscow Power Engineering Institute, Russia, etc. His current research targets fundamental algorithms for computer graphics, geometric algebra, meshless (meshfree) methods for scalar and vector fields interpolation and approximation, and applied mathematics.

He is the Editor-in-Chief of the Journal of WSCG and Computer Science Research Notes (CSRN).

Vaclav Skala has published over 160+ research-indexed papers with more than 1000+(WoS/Publons), 1570+(Scopus), and 3000+(Scholar Google) citations.

A CNN-RNN Hybrid Approach for Polish License Plate Recognition: Harnessing Transfer Learning and Real-World Validation

Gergő B. Békési* and Péter Ekler
Department of Automation and Applied Informatics,
Faculty of Electrical Engineering and Informatics,
Budapest University of Technology and Economics, Budapest, Hungary
**Corresponding author: Gergő B. Békési (bekesi.gergo@aut.bme.hu)*

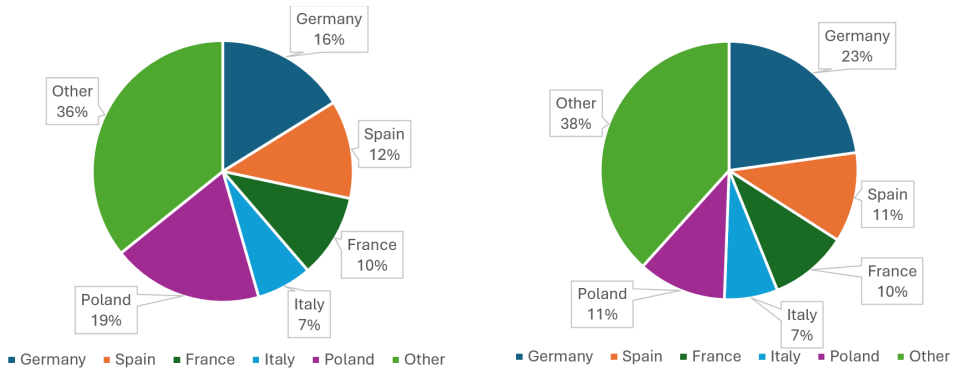
Submitted: Jan 1, 2025 **Accepted:** May 15, 2025 **Published:** Jun 23, 2025

Abstract Automated license plate recognition (LPR) systems have garnered substantial attention within the field of intelligent transportation systems, owing to their pivotal role in facilitating toll collection, enhancing traffic management, and ensuring operational efficiency. Despite recent breakthroughs in convolutional and recurrent neural network architectures, Polish LPR remains underexplored, with most existing approaches relying on conventional optical character recognition. This study proposes a hybrid convolutional neural network – recurrent neural network (CNN-RNN) model equipped with a Thin-Plate Spline (TPS) transformation module, a ResNet-based feature extractor, a bidirectional Long Short-Term Memory (LSTM) sequence model, and an attention-based decoder to address the unique challenges of Polish license plates. The model is trained on a high-difficulty dataset, comprising real-world images without explicit character-level bounding boxes. Empirical evaluations underscore the efficacy of the proposed system, with competitive accuracy and normalized edit distance scores achieved on Polish, Czech, Hungarian, and Slovak datasets. Additionally, transfer learning from closely related Central European plate formats to Polish data demonstrates marked improvements in convergence and overall performance. Further validation on a challenging video-based dataset reveals the robustness of the proposed approach, evidencing its potential applicability in real-world scenarios and highlighting majority voting as an effective strategy to enhance system reliability under variable conditions.

Keywords: Polish license plate recognition, CNN-RNN, transfer learning, majority voting.

1. Introduction

Automated systems significantly enhance operational efficiency, mitigate human error, and facilitate expedited decision-making processes. Within the domain of transportation engineering, active traffic management strategies employ real-time data acquisition and predictive algorithms to optimise roadway performance and alleviate congestion through the dynamic regulation of traffic flow, particularly during peak hours or in response to incidents [30]. Intelligent transportation systems integrate advanced communication and sensor technologies to enhance traffic operations. These systems enable functionalities such as vehicle-to-infrastructure communication and centralized monitoring, thereby reducing delays and improving overall safety [1]. Automated lane control managed by centralized lane coordination mechanisms, facilitate the adaptive allocation of road lanes based on traffic demand. This approach enhances roadway capacity and



a Distribution in million vehicle-kilometres. **b** Distribution in thousand basic transport operations.

Fig. 1. Road freight transport by distance in the EU, Norway, and Switzerland [10].

mitigates bottlenecks during periods of high demand [23]. Furthermore, dynamic traffic flow optimisation utilises sophisticated modelling techniques and real-time data inputs to adjust parameters such as speed limits and lane assignments, thereby minimizing travel times and reducing fuel consumption [50]. Traffic incident management systems leverage automated alert technologies and resource coordination to enable prompt responses to roadway incidents, thus reducing disruptions and improving traffic safety [24]. Adaptive traffic signal control systems, employing machine learning algorithms, dynamically adjust signal timings to accommodate real-time traffic conditions, effectively reducing delays and emissions [44].

The exponential growth of global transportation highlights the pivotal role of automation in state-managed road infrastructure. This extends to freight transportation, where automated toll collection systems, leveraging license plate recognition technology, ensure precise and reliable monitoring [13]. These systems are integral to optimising operational efficiency and sustaining the economic viability of road networks amidst rising transportation demands.

Within the European Union, freight transportation, as measured in vehicle-kilometres, demonstrates significant variability among member states. As depicted in Figure 1, Poland ranks highest in vehicle-kilometres for freight transport and is also dominant in the number of basic transport operations, underscoring its pivotal role within the European logistics network.

Poland's strategic geographic location and well-developed infrastructure position it as a vital transit hub, facilitating connections between Eastern and Western Europe. The dataset also includes Norway and Switzerland, which, while not EU members, maintain close economic ties with the Union. Notably, no data for Liechtenstein was available in the Eurostat database [10]. Therefore, the implementation of efficient LPR systems is

particularly critical in Poland, as these systems play a vital role in ensuring the seamless management of transit traffic and optimising the utilisation of road infrastructure within the European logistics network.

The structure of the remainder of this paper is as follows. In Section 2, an overview is provided of state-of-the-art methodologies for LPR, analysing the unique challenges associated with recognising Polish license plates and positioning Polish LPR-focused projects within the broader context of related research. Section 3 introduces an approach to address a gap in the Polish LPR literature by integrating RNNs with CNNs. It also describes the dataset used in this study, characterized by its high level of difficulty, and assesses the efficacy of transfer learning in applying machine learning models trained on license plate datasets from other countries. Section 4 presents the experimental results, while Section 5 concludes the study by summarising the key findings and their implications.

2. LPR and Polish LPR

Polish LPR differs from the LPR systems of other nations not only in terms of license plate format but also in the technologies employed. In this section, the state-of-the-art is introduced, the unique features of the Polish license plate format are highlighted, and the characteristics of Polish LPR projects are described.

2.1. State-of-the-art in LPR

Over the past decade, LPR has evolved from relying on handcrafted features and rule-based heuristics – such as edge detection [35], morphological operations [49], and character segmentation using thresholds [20] – to employing deeply learned representations implemented through various neural network architectures.

While handcrafted methods relied heavily on prior knowledge and manually tuned parameters to detect and recognise license plates under constrained conditions, deep learning-based techniques learn hierarchical features directly from data, enabling improved robustness and generalization across varying environments.

Contemporary LPR solutions can generally be categorized into three main deep learning paradigms: CNN-based pipelines [39], RNN-based frameworks [31], and single-shot detection methods, such as those built on You Only Look Once (YOLO) [2] or Single Shot Detector (SSD) [40] architectures. Each approach entails distinct trade-offs in terms of model complexity, speed, accuracy, and annotation requirements [22].

CNN-based LPR solutions typically follow a pipeline in which character recognition occurs after an initial license plate detection stage. Early methods in this category often employed separate subsystems: one CNN for plate detection, another for character segmentation, and a final CNN for character classification. This step-by-step strategy

first localizes the plate in the image, then segments individual characters, and finally classifies each character independently [39].

These CNN architectures leverage learned visual features extracted directly from raw pixel data. They are proficient in accommodating variations in character fonts, plate backgrounds, and moderate viewpoint changes. However, a critical limitation is their reliance on explicit character segmentation. This segmentation step usually requires tightly bounded boxes or heuristics to isolate each character before classification, thereby increasing data annotation complexity and the risk of segmentation errors [33]. Moreover, while CNN-based methods have demonstrated state-of-the-art accuracy under controlled conditions, they can falter when facing substantial geometric distortions or highly non-uniform illumination.

RNN-based LPR architectures are designed to overcome some of the inherent limitations of purely CNN-driven pipelines, particularly regarding character segmentation [7]. Instead of explicitly segmenting characters, RNN-based methods conceptualize the recognition task as a sequence labelling problem. The network ingests either raw plate images or CNN-extracted features and generates a sequence of predicted characters. This approach eliminates the need for character-level bounding boxes since only a correct global sequence label is required [31]. As a result, annotation becomes considerably simpler, as labelling the entire license plate string suffices.

RNN-based models effectively handle variations in character length and spacing while preserving contextual dependencies between characters [16]. This leads to enhanced robustness, particularly for plates with irregular spacing or slight rotations. However, RNN-based solutions may introduce a higher computational overhead, and their performance can depend on the quality of the initial features, often provided by a preceding CNN. Despite these computational costs, these models gracefully circumvent the segmentation challenge, frequently improving robustness and reducing manual annotation efforts [47].

Single-shot detectors, including YOLO and SSD, represent another prominent approach to LPR. Unlike multi-stage pipelines, single-shot methods attempt detection and, in some cases, recognition within a single feed-forward pass [5]. YOLO-based architectures divide the image into a grid and directly predict bounding boxes and class probabilities [2], thereby achieving remarkable inference speed. SSD similarly applies a single forward pass to detect objects at multiple scales, which can be advantageous for handling plates of varying sizes and positions [40].

A key advantage of single-shot methods is their computational efficiency, enabling near real-time recognition on standard GPU hardware. Their speed is appealing for real-world applications requiring high throughput, such as toll booths, traffic monitoring, or large-scale surveillance systems. However, these models can struggle with very small plates, especially when the field of view is extensive, or when plate distortion is

significant. Although single-shot methods can be adapted to handle character recognition, this often involves treating each character as a separate object [27]. Without careful calibration or sophisticated refinement steps, their accuracy may not match that of specialised CNN- or RNN-based solutions, particularly under challenging conditions.

CNN-based and RNN-based approaches can produce highly accurate results but may be slower due to multi-stage processing or the overhead associated with recurrent computations. In contrast, single-shot models deliver improved speed, which is invaluable for real-time applications, albeit potentially at the expense of slightly lower accuracy in complex scenarios. Furthermore, CNN-driven and single-shot pipelines typically require bounding boxes for both plates and individual characters, whereas RNN-based methods greatly simplify annotation by relying solely on the license plate's textual content. All three approaches address variations in plate appearance, lighting, and orientation. CNN-based methods depend on pre-processing and carefully engineered pipelines, RNN-based methods accommodate irregular spacing and character sequences without explicit segmentation, and single-shot detectors excel at rapid processing but may require meticulous tuning to handle small plates and severe distortions effectively.

The choice among CNN-based, RNN-based, or single-shot solutions ultimately depends on the priorities of the application. CNN architectures remain strong candidates when high accuracy is paramount and per-character bounding box annotations are readily available. RNN-based approaches simplify labelling and segmentation tasks, rendering them highly attractive in diverse and distorted plate conditions. YOLO and SSD models stand out in scenarios where speed is critical, even if doing so involves a trade-off against absolute accuracy.

2.2. Polish LPR

Polish license plates adhere to the standardized European format while incorporating unique features defined by national regulations. This subsection provides an overview of the Polish license plate standard and analyses how Polish LPR systems diverge from the global state-of-the-art approaches.

2.2.1. Polish License Plate Format

Polish vehicle registration plates are standardized under national regulations, ensuring uniformity and comprehensive traceability throughout the country. Contemporary registration plates comply with the European Union's standardized aesthetic guidelines, featuring black alphanumeric characters embossed on a white, reflective background, along with a blue European stripe displaying the "PL" country code and the EU stars. This harmonized design has been adopted to enhance visibility, interoperability, and consistency within the broader European context.

A hierarchical geographic encoding scheme underpins the Polish plate system. Each plate's prefix encodes its origin at both the voivodeship (province) and county levels.

Fig. 2. A Polish license plate from the dataset.

Specifically, the initial letter denotes the voivodeship, while subsequent letters specify the county. Additional letters may follow to represent larger urban centres or historically significant administrative divisions. These area codes are succeeded by a series of digits, and in some instances, additional letters, forming a unique alphanumeric identifier for each vehicle. To address the diminishing pool of available combinations in densely populated regions, supplementary leading characters have been introduced, thus expanding the combinatory space without necessitating a comprehensive overhaul of the national registration format. A Polish license plate, illustrating this hierarchical geographic encoding scheme, is presented in Figure 2.

In addition to standard passenger vehicle plates, a range of specialised categories is employed to accommodate varying vehicle types and operational contexts. Different configurations and dimensions are used for motorcycles, mopeds, and agricultural vehicles, often employing smaller, two-line layouts. Reduced-size plates exist for vehicles imported from the United States and Japan to ensure proper fit. Moreover, vehicles utilising electric or hydrogen propulsion are issued plates with a green background, a feature that aids in regulating privileges such as bus lane access. Other specialised variants—such as those for historical cars, temporary registrations, vehicle testing, professional dealerships, diplomatic missions, military units, the police, and other government services—adhere to specific colour schemes, character arrangements, and symbolic elements. In doing so, the system accommodates diverse transportation needs while maintaining a coherent and functionally robust identification framework [34].

2.2.2. Polish LPR Projects

Despite the widespread adoption of CNNs, single-shot methods, and CNN-RNN hybrid approaches in contemporary LPR systems, the domain of Polish LPR exhibits a distinct reliance on conventional OCR-based methodologies. While CNNs are occasionally integrated within OCR pipelines, primarily for character recognition tasks, and single-shot detection frameworks are sporadically employed, the utilisation of RNNs remains notably absent across the reviewed Polish LPR projects.

Polish researchers have demonstrated notable successes in various computer vision applications—ranging from object detection [45] and face recognition [32] to medical image analysis [26] and aerial surveillance [12]. Nevertheless, these achievements have not been systematically extended to the LPR domain, perhaps due to the long-standing availability of commercial and open-source OCR-based solutions. Consequently, large-scale,

specialized approaches for license plate recognition have thus far received comparatively less attention. This subsection reviews the existing works, illustrating how classical segmentation and recognition pipelines dominate the Polish domain and highlighting the conspicuous absence of RNN-based methods.

Kluwak et al. [25] employ multi-frame recognition and object tracking to enhance single-frame license plate recognition. Although the authors introduce a novel mechanism for combining results from consecutive video frames, known as In Track Clustering Correction (ITCC), their approach relies on established computer vision techniques through the default settings of OpenALPR [38], which is a computer vision based software for LPR tasks written in C++. While the authors do not provide a detailed description of preprocessing, one may infer that it involves binarization and filtering to address noise under adverse weather conditions, followed by morphological operations such as dilation and erosion and subsequent histogram transformations. These steps likely facilitate character segmentation through the detection of connected components, often referred to as tracks of recognised characters, and are then followed by CNN-based character recognition. Subsequently, the outputs of conventional optical character recognition are refined through a clustering-based voting process across frames. Although alternative strategies could be employed for single-frame LPR in the presented pipeline, the authors continue to use these standard methods in their implementation.

Janowski et al. [18] investigate the effectiveness of LPR in video streamed under constrained networking conditions, focusing on both human and automatic recognition systems. The study examines two custom-built LPR algorithms, the Labelling and Artificial Neural Networks (LANN) and the Periodic Walsh Piecewise-Linear Descriptors (PWPLD). The LANN algorithm follows a traditional pipeline, involving preprocessing with Otsu binarization and morphological operations, license plate localization through connected component analysis, segmentation for character extraction, and recognition via a neural network trained on scaled character images. The PWPLD algorithm incorporates preprocessing with contour-based features, license plate detection using brightness regularities, and recognition through decision-tree rules generated by See5 [41] software. Both methods were evaluated against human recognition, demonstrating that while human accuracy exceeded that of the algorithms, the LANN method performed comparably under certain conditions aided by adaptive feedback mechanisms and parameter tuning. The authors measured performance under both compressed conditions (CC) and no compression (NC), using accuracy and the $(1 + \mathrm{LED})^{-1}$ as evaluation measures, where LED is the Levenshtein distance [29] sum to character number sum ratio.

Kekez [21] proposes a multi-stage approach for real-time LPR that utilises a cascade of classifiers to achieve high accuracy while minimizing training data requirements. The process begins with preprocessing the license plate image, which includes greyscale conversion, binarization, negation, and morphological operations to enhance the plate's features. Character segmentation is performed using connected component analysis,

with optional dilation or erosion to address challenges such as poor lighting or occlusions. The character recognition stage employs a cascade of classifiers, starting with a neural network for initial classification, followed by parallel Random Forest, decision tree, or rule-based classifiers to refine results. A final stage applies contextual rules to improve accuracy further, such as disambiguating similar characters based on their position within the plate. The proposed system achieves over 98% recognition accuracy by leveraging small, country-specific training datasets derived from legal specifications rather than real license plate images, offering a computationally efficient alternative to deep learning methods like YOLO, which the author also highlights as an important architecture in the field.

Leszczuk et al. [28] propose an objective evaluation methodology for assessing video quality tailored to automatic LPR under various challenging conditions. The approach integrates a dataset of video frames annotated with ground truth coordinates, simulating real-world scenarios such as occlusions, motion blur, and lighting variations. License plate recognition is performed using the aforementioned OpenALPR library, which combines traditional image processing techniques with OCR to detect and classify license plates. The recognition pipeline utilises confidence scores and country-specific settings to improve accuracy. The study highlights the influence of video distortions on LPR performance, such as Gaussian noise, defocus, and JPEG compression, and demonstrates the limitations of standard video quality metrics in recognition tasks. The methodology achieves an F-measure of 0.777, underscoring its potential as a robust framework for evaluating and enhancing video-based LPR systems in diverse operational conditions.

Wroblewski [46] introduces a real-time algorithm for recognising Polish car license plates, achieving a recognition rate of 60-70%. The method follows a three-stage process: plate localization, symbol segmentation, and symbol recognition. Plate localization involves detecting and isolating the license plate area from the input image using a modified edge detection algorithm that emphasizes vertical edges, as these are prominent in license plate regions. Symbol segmentation extracts individual characters from the plate image, while recognition is performed using a neural network-based approach. The algorithm is designed for cost efficiency, requiring only the computational power of a personal computer.

Łubkowski and Laskowski [52] investigate the quality and reliability of automatic LPR systems in the context of diverse environmental and technical conditions. Their analysis focuses on critical factors such as light intensity, image resolution, and camera angles, and how these influence the performance of license plate detection and recognition. The automatic LPR workflow encompasses image acquisition, localization of the license plate, segmentation of individual characters, and their subsequent recognition, facilitated by optical character recognition and syntactic validation algorithms. The researchers employed the Simple-LPR application in a controlled experimental setup, testing its functionality under varying scenarios. The results reveal that the system achieves reliable

recognition under favourable lighting conditions, specifically light intensities ranging from 5000 to 7000 lux, and at standard image resolutions. However, when using low-resolution images performance declines significantly.

Jureczko and Uherek [19] propose a video-based automatic LPR system utilising CNNs for accurate and efficient character recognition. The system employs Tiny YOLO for vehicle detection and localization within video frames, isolating regions of interest (ROIs) containing license plates. These ROIs undergo image processing steps, including Gaussian blur, thresholding, and contour detection using OpenCV [4], to extract license plates. Character recognition is performed with the Tesseract OCR [42] library, enhanced by multiple thresholding iterations and a voting mechanism to ensure consistent results. To optimise processing, a dictionary-based structure tracks previously recognised plates, avoiding redundant computations. Tests demonstrated the system's ability to accurately recognise license plates in controlled lighting conditions, processing frames at 0.62 seconds per frame. While effective in stable environments, the system's performance is hindered by adverse weather conditions and inconsistent lighting, highlighting the need for further improvements for robust, real-world applications.

The reviewed projects highlight a predominant reliance on OCR-based solutions and a notable absence of RNN-based approaches in Polish LPR systems. This study aims to introduce a CNN-RNN-based solution while acknowledging that current Polish LPR solutions primarily focus on OCR. The proposed model is designed to be potentially integrable into widely popular frameworks such as the EasyOCR pipeline [17]. Nonetheless, the central focus of this research is the training of the CNN-RNN model. It aims to contribute an original Polish LPR solution anchored in a CNN-RNN approach, thereby addressing the notable gap in current research and paving the way for further advancements in accurate, efficient, and versatile license plate recognition for Poland. Additionally, this study investigates how models trained on non-Polish license plates can be effectively fine-tuned to adapt to Polish license plates, emphasizing the importance of adaptation and transfer learning in this domain.

3. Methods and data

In the subsequent subsections, we present the overall approach for the proposed Polish LPR system, beginning with a detailed explanation of the hybrid CNN-RNN method. This covers the role of a Thin-Plate Spline transformation, a ResNet-based feature extractor, and an attention-based decoder within the recognition pipeline. We then introduce the data used to train and validate our model, emphasizing the challenges inherent in real-world license plate images and the reasons behind choosing the data resource.

3.1. Methods

A typical CNN-RNN approach for text recognition, including license plate recognition, typically involves three primary stages: feature extraction, sequence modelling, and prediction [51]. In certain scenarios, an additional preliminary transformation stage may be incorporated before feature extraction to address geometric distortions. When significant geometric variations such as curved text or perspective distortions are anticipated, researchers often include this optional transformation stage at the beginning of the pipeline. Methods such as Spatial Transformer Networks are frequently employed to rectify input images, thereby simplifying subsequent processing and enhancing overall recognition performance [37].

In the standard architecture, regardless of whether the transformation stage is included, a CNN is initially used to extract high-level features that minimize irrelevant elements such as background clutter or font variations. These features are then processed by an RNN, typically a Gated Recurrent Unit (GRU) [9], a LSTM network [43], or a bidirectional LSTM [15], to capture contextual relationships between characters within the textual sequence, such as the structural patterns of a license plate. Finally, the text is recognised using a decoder, which may utilise either Connectionist Temporal Classification (CTC) [8] to align input frames with output labels through a collapsing strategy or attention-based mechanisms that dynamically focus on specific segments of the feature sequence.

In the present work, we employ a variant of the CNN-RNN-based pipeline [3] that leverages a TPS transformation module, a ResNet-based [14] feature extractor, a bidirectional Long Short-Term Memory (BiLSTM) [11] sequence model, and an attention-based decoder to recognise license plates under various geometric and visual distortions. Mathematically, let $\mathbf{x} \in \mathbb{R}^{H \times W}$ denote the input license plate image, represented as a single-channel grayscale image of height H and width W. Our approach first applies a TPS Spatial Transformer Network (TPS-STN) that learns a parametric transformation to rectify distorted inputs. We define a set of F fiducial control points $(C_i^x, C_i^y)_{i=1}^F$ in normalized coordinates on the input image. The TPS transformation warps $\mathbf{x}$ to a canonical form $\mathbf{x}' \in \mathbb{R}^{H' \times W'}$ via the mapping:

$$T_{\text{TPS}} : (u, v) \mapsto (u', v') = \phi\left((u, v) \mid \theta\right), \tag{1}$$

where ϕ is obtained by solving a system of radial basis functions centered at the fiducial points, and θ encapsulates the learnable parameters of the localization sub-network. By aligning the text region into a more "straightened" coordinate frame, the TPS module simplifies subsequent recognition.

Once the image is rectified, it is passed to the ResNet-based feature extractor. Denoting the rectified image as $\mathbf{x}'$, a series of convolutional and downsampling layers $f_{\text{ResNet}}(\cdot)$

yield a deep feature map:

$$\mathbf{F} = f_{\text{ResNet}}(\mathbf{x}') \in \mathbb{R}^{H_f \times W_f \times C_f}, \tag{2}$$

where H_f, W_f, and C_f respectively denote the height, width, and number of channels in the final feature representation. The ResNet backbone is composed of stacked residual blocks, which improve gradient flow and mitigate vanishing gradients, facilitating feature learning even in complex visual environments such as license plates with varied fonts or cluttered backgrounds.

Next, we reshape the spatial dimensions $H_f \times W_f$ of $\mathbf{F}$ into a sequence of feature vectors along the width dimension:

$$\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_T = \text{reshape}(\mathbf{F}), \tag{3}$$

where $T = H_f \times W_f$. These vectors are then fed into a bidirectional LSTM:

$$\mathbf{h}_t = \text{BiLSTM}(\mathbf{f}_t, \mathbf{h}_{t-1}), \tag{4}$$

where $\mathbf{h}_t \in \mathbb{R}^d$ is the hidden state at step t, combining both forward and backward passes. The LSTM gating mechanisms selectively retain or discard information, allowing the model to capture dependencies among character regions. The bidirectionality is particularly useful for recognizing alphanumeric patterns, as it captures context from both directions.

Finally, an attention-based decoder predicts the output character sequence. Denote the full set of BiLSTM hidden states as $\mathbf{H} = \mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_T$. At decoding step k, the attention mechanism computes a context vector $\mathbf{c}_k$ as:

$$\alpha_{k,t} = \text{softmax}\left(\mathbf{w}^\top \tanh\left(\mathbf{W}_h \mathbf{h}_t + \mathbf{W}_s \mathbf{s}_{k-1}\right)\right), \tag{5}$$

$$\mathbf{c}_k = \sum \alpha_{k,t} \mathbf{h}_t, \tag{6}$$

where $\alpha_{k,t} \in [0, 1]$ are the attention weights, $\mathbf{s}_{k-1}$ is the previous decoder state, and $\mathbf{w}$, $\mathbf{W}_h$, and $\mathbf{W}_s$ are learnable parameters. The decoder state then updates via:

$$\mathbf{s}_k = \text{LSTMCell}(\mathbf{c}_k, \mathbf{s}_{k-1}). \tag{7}$$

producing the distribution over possible next characters. By focusing on the most relevant portions of the feature sequence at each decoding step, attention avoids the assumption of rigid alignment, as in standard CTC, and is more robust to variable character spacing, occlusions, or partial distortions frequently found in license plate scenarios. The block diagram of the proposed architecture is illustrated in Figure 3.

This architecture was trained on Central European license plates, specifically from Poland, as well as from other Visegrád Group countries (Czech Republic, Hungary,

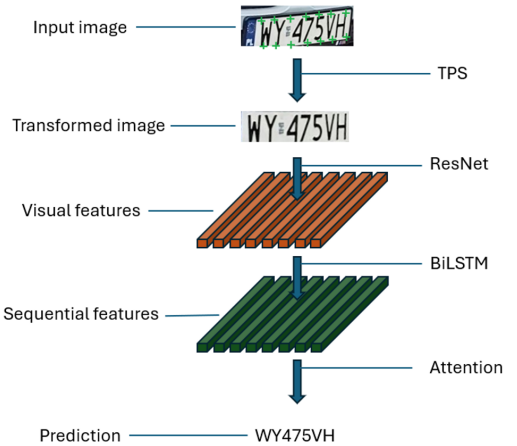


Fig. 3. Block diagram of the proposed architecture.

and Slovakia). The aim was to investigate whether transfer learning, applied to models initially trained on license plates from some geographically proximate countries, enhances performance. The characteristics of the dataset are detailed in the subsequent subsection.

The initial training phase consisted of 50 epochs using the Adadelta [48] optimisation algorithm. Adadelta was selected for this text recognition task due to its adaptive learning rate properties, which are particularly effective in scenarios where feature scaling and regularization are critical. A starting learning rate of 1.0 was employed, with a decay rate of 0.95 to ensure gradual adjustment of the learning rate during training. The epsilon parameter of Adadelta, set to 10^{-8}, played a key role in maintaining numerical stability, especially when updating the learning rate for sparse gradient scenarios. Gradient clipping was applied with a threshold of 5, which helps prevent exploding gradients, thereby stabilizing the training process.

Following the base training, a transfer learning phase, referred to here as retraining, was conducted for 30 epochs using the same hyperparameters. This phase started with the models trained on the specific datasets from the Czech Republic, Hungary, and Slovakia, and continued by retraining them on the Polish dataset. This approach aimed to refine the model's performance by leveraging domain-specific features from the respective datasets.

To ensure practical applicability in real-world scenarios, the developed models are designed to operate sequentially over temporally ordered image frames, such as those extracted from video streams. This temporal deployment capability allows the recognition pipeline to process each frame individually, yielding a predicted text output per frame. In the present study, this approach was employed during the testing phase:

after the transfer learning stage, the best-performing model – selected based on its validation accuracy – was evaluated on video sequences from the aforementioned Parking Database [28]. For each frame in a sequence, the model predicted a license plate text; the final recognised plate was then determined by selecting the most frequently predicted string across all frames (a strategy commonly referred to as majority voting, where the mode of individual predictions is taken as the final decision). This strategy enhances robustness against frame-level noise, motion blur, and partial occlusions, which are common in surveillance footage. The effectiveness of this method is illustrated later in Subsection 4.4.

The CNN-RNN architecture offers strong capabilities for sequential text recognition; its novelty in this work lies in its application to the recognition of Polish license plates – a task that has received limited attention in prior research. By embedding this architecture within a majority voting framework for video-based inference, the method demonstrates practical robustness in dynamic, real-world scenarios.

3.2. Data

The dataset utilised in this study originates from a platform dedicated to vehicle license plates and related topics, including their history, production technologies, usage, and statistical distribution. It serves as an extensive online repository where users can upload and share images of vehicles and license plates. The platform fosters an engaged community of enthusiasts who discuss vehicle characteristics, evaluate uploaded images, and exchange observations. Although license plates can be searched based on their alphanumeric identifiers, the website adheres to strict privacy policies and does not associate license plates with personal information about vehicle owners.

The dataset employed in this research specifically includes license plate images from Poland and the countries examined for transfer learning and was curated from publicly accessible resources. The images feature a wide range of real-world scenarios, including variations in lighting conditions, viewing angles, and plate appearances. Notably, this dataset does not include character-level bounding boxes or individual character annotations. Instead, each license plate is annotated with a single text label representing the complete license plate string. This characteristic aligns well with the requirements of RNN-based architectures, which excel in sequence labelling tasks by processing the entire plate string as a single sequence rather than relying on explicit character segmentation.

The absence of character-level bounding boxes, while simplifying annotation efforts, presents unique challenges for automated LPR systems. These challenges are particularly pronounced in datasets with diverse plate designs, irregular spacing, and distortions. Consequently, this dataset is considered more demanding than those providing detailed character-level annotations. However, its characteristics render it highly suitable for RNN-based methodologies, which inherently bypass the segmentation requirement by focusing on sequence-level recognition.

Fig. 4. Examples of the images from the dataset.

The utilisation of this dataset supports the development of Polish CNN-RNN hybrid models, emphasizing sequence labelling as a robust approach to LPR. Furthermore, leveraging this dataset underscores the importance of adapting algorithms to real-world, imperfect conditions, enhancing the applicability and generalizability of the resulting models to practical scenarios. For each of the four countries under study—Poland, the Czech Republic, Hungary, and Slovakia—exactly 1000 images were available, which were divided into a training-validation split of 800 and 200 images, respectively. Although the data source hosts a substantially larger number of license plate images overall, it only provides up to 1,000 images per country in publicly accessible form. Some examples from the dataset, illustrating its diversity and relevance to the proposed approach, are presented in Figure 4.

In designing the training set, special consideration was given to the distinct format and characteristics of Polish license plates, which differ from those of the other countries. For instance, Polish plates often feature regional codes encoded in their prefixes, and the overall alphanumeric structure may include unique character arrangements not found elsewhere. To capture these idiosyncrasies, the Polish subset was curated to include diverse plate designs (e.g., specialized plates for electric vehicles, two-line motorcycle plates, and regional variations). These differences also guided architectural decisions—most notably, the use of a sequence-based approach that can handle variable text lengths and character sets. By explicitly accounting for these features, the model is better equipped to recognise Polish plates accurately while still leveraging cross-country similarities for transfer learning and overall robustness.

While Polish license plates share certain similarities with other European formats – such as Czech plates, which also encode geographic information in the initial characters – no large-scale quantitative analysis has yet been devoted specifically to Polish LPR, unlike, for instance, the Czech case [36]. Consequently, this study addresses an important gap by investigating how domain knowledge from related Central European plate formats can be leveraged via transfer learning. Moreover, the CNN-RNN architecture is particularly well-suited for capturing positional dependencies, including region codes often found at the beginning of Polish plates. By examining how effectively these sequence-based features can enhance Polish LPR, the study underscores both the unique

Fig. 5. Realistic challenges of the dataset.

aspects of Polish license plates and the broader applicability of such a model for other European systems with similar encoding schemes.

In addition to training and evaluating Polish CNN-RNN hybrid models, this dataset was also used to test the best-performing architecture through majority voting on video sequences from the Parking Database, consisting of 22 videos recorded by parking lot cameras. The Parking Database videos, freely available to the research community, provide challenging real-world conditions for evaluating license plate recognition models. Each video includes corresponding ground truth annotations and vehicle registration plates as reference data. This video dataset was accessed directly without applying any preprocessing steps or distortion models as outlined in Leszczuk et al. [28].

Moreover, it must be noted that there are realistic images in the training and evaluation set, not just in the test set. Figure 5 showcases some of the more challenging license plate images within our dataset, captured under suboptimal lighting, precipitation, or partial occlusion. These conditions reflect real-world limitations that can hinder accurate recognition and are therefore crucial to include for robust model training and evaluation. By incorporating examples of low-quality plates, our dataset ensures that the proposed approach is tested against a broad spectrum of difficulties. Furthermore, using majority voting over consecutive frames helps mitigate transient noise from these adverse conditions, improving overall recognition stability.

4. Results

This section presents the evaluation of the proposed CNN-RNN-based LPR architecture. First, the performance is assessed for models trained on Polish, Czech, Hungarian, and Slovak datasets over 50 epochs in Subsection 4.1. Two measures are used: accuracy – the percentage of license plates fully correctly predicted – and normalized edit distance (normED) – which measures similarity between the predicted and ground-truth plate strings. Next, the effects of retraining are investigated by adapting models initially trained on Czech, Hungarian, and Slovak license plates to Polish data in Subsection 4.2.

Following that the robustness to weather conditions is examined in Subsection 4.3. Finally, Subsection 4.4 reports the results of testing on real-world video sequences, where majority voting is applied to enhance recognition stability.

4.1. Baseline training on individual datasets

Each of the four baseline models was trained for 50 epochs on its corresponding national dataset (Polish, Czech, Hungarian, and Slovak). Let N denote the total number of license plates in the validation set. The accuracy measure is defined as:

$$\text{Acc} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}(\hat{y}_i = y_i), \tag{8}$$

where $\hat{y}_i$ represents the predicted license plate for the i-th sample, y_i is the corresponding ground-truth label, and $\mathbb{I}(\cdot)$ is the indicator function that returns 1 if the predicted label matches the ground truth exactly, and 0 otherwise.

To capture partial string mismatches, the normalized edit distance is employed:

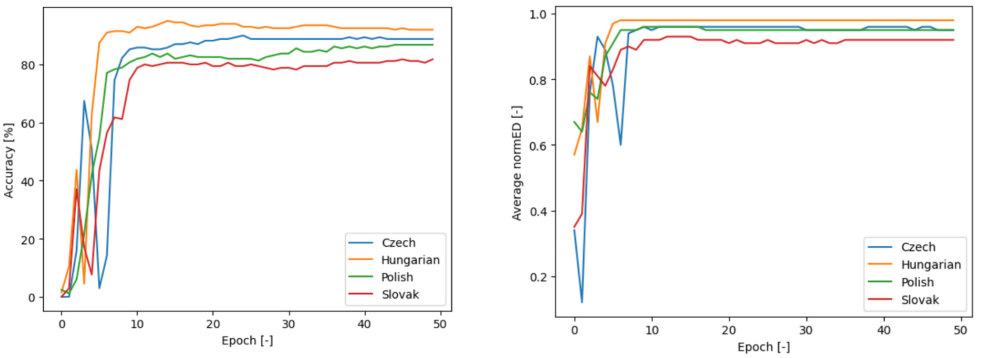
$$\text{normED} = 1 - \frac{d_{\text{lev}}(\hat{y}_i, y_i)}{\max(|\hat{y}_i|, |y_i|)}, \tag{9}$$

where $d_{\text{lev}}(\hat{y}_i, y_i)$ is the Levenshtein distance between the predicted and ground-truth strings, and $|\cdot|$ denotes string length. Higher normED values indicate greater similarity between $\hat{y}_i$ and y_i.

These two evaluation metrics were selected to jointly reflect the dual nature of the task, which lies at the intersection of pattern recognition and text recognition. Accuracy, a standard metric in the pattern recognition domain, quantifies the proportion of completely correct predictions and is widely used in the literature, facilitating comparison not only with prior work in this domain but also across different application areas. Its inclusion enables us to position the difficulty of the task within a broader machine learning context.

However, in the specific case of license plate recognition, where predictions are structured character sequences, exact-match accuracy alone may not sufficiently capture the quality of near-correct outputs. Therefore, normED, a well-established metric in the text recognition domain, is also adopted. Unlike simple character-level accuracy, normED penalizes both over- and under-predictions, and—by normalizing the Levenshtein distance—enables fair evaluation across license plates of varying lengths. This makes it particularly suitable for sequence-based recognition tasks, providing a more nuanced and stringent assessment of prediction errors.

It is also worth noting that, in the context of Polish license plate recognition, earlier studies have employed alternative text-level metrics such as the inverse of (1 + LED)



a The accuracy curves.

b The normED curves.

Fig. 6. Training curves for baseline training.

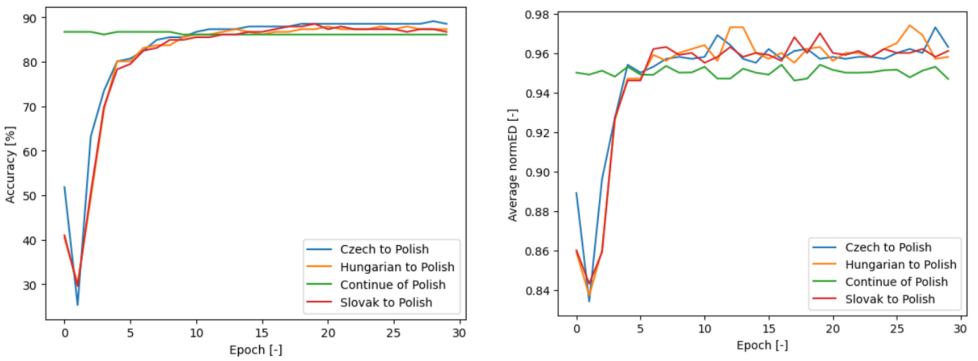
and character-level accuracy, as discussed in Subsection 2.2.2. To ensure comparability with these prior works, we computed these metrics for our models as well, and the corresponding results are presented in Subsection 4.2.

Figures 6a and 6b show the accuracy and average normED curves, respectively, over 50 epochs for the four baseline models. From Figure 6a it results that the Hungarian dataset achieves the highest final accuracy (95.0%), followed by the Czech (89.9%), Polish (87.3%), and Slovak (84.1%) datasets. The Hungarian model also displays relatively steady convergence. In contrast, the Czech accuracy curve undergoes more pronounced fluctuations in the early epochs before stabilizing. The curves for the Polish and Slovak sets are not as smooth as the one for Hungarian data but exhibit fewer fluctuations than the Czech curve overall.

As shown in Figure 6b, the normED measure follows a similar pattern. The Hungarian model reaches approximately 98.3%, the Czech model 96.0%, the Polish model 95.8%, and the Slovak model 92.3%. Notably, the differences in normED across the four countries are smaller than those observed for accuracy, indicating that even when the entire license plate is not predicted perfectly, the predicted string remains relatively close to the ground truth. Overall, the Polish model consistently ranks near the middle of the performance range and demonstrates stable, incremental improvements throughout training.

4.2. Transfer learning

A transfer learning phase was conducted to examine whether knowledge learned from one Central European dataset could be transferred to Polish license plate recognition. Specifically, the final weights of three baseline models (Czech, Hungarian, and Slovak)



a The accuracy curves.

b The normED curves.

Fig. 7. The curves of retraining.

were used as initialization for Polish LPR training. Each retraining session ran for an additional 30 epochs under the same hyperparameter settings.

Figures 7a and 7b depict the accuracy and normED curves, respectively, for these retrained models. There is also a Continue of Polish curve included to illustrate the effect of extending training solely on the Polish dataset without external initialization. This provides a direct comparison between simple prolonged training and transfer learning, highlighting the relative benefits of incorporating prior knowledge from related domains.

All three retraining curves in Figure 7a exhibit smoother convergence compared to their respective baselines. The continuous Polish training does not show a significant change in terms of accuracy. The final accuracy values after 30 epochs are:

- Czech-based retraining: 89.2%,
- Hungarian-based retraining: 88.0%,
- Continued training on Polish: 87.2%,
- Slovak-based retraining: 88.6%.

A similar trend is observed in Figure 7b, although with a slight decrease in the Polish-only case for the normED measure. The final normED values are:

- Czech-based retraining: 97.3%,
- Hungarian-based retraining: 97.4%,
- Continued training on Polish: 94.7%,
- Slovak-based retraining: 97.0%.

So, these are the last values of the series of epochs. Importantly, continuing training on the Polish dataset for an additional 30 epochs without introducing new knowledge led to a slight decline in both validation accuracy and normED (from 87.3% to 87.2% and from 95.8% to 94.7%, respectively). This suggests that further training on the

Tab. 1. Performance before and after transfer learning on Polish license plate data.

Data	Accuracy	Accuracy increment	normED	normED increment
Continued training on Polish	87.2%	−0.1%	94.7%	−1.1%
Czech to Polish	89.2%	+1.9%	97.3%	+1.5%
Hungarian to Polish	88.0%	+0.7%	97.4%	+1.6%
Slovak to Polish	88.6%	+1.3%	97.0%	+1.2%

same dataset may induce overfitting, whereby the model becomes too specialized to the training data and fails to generalize better to the validation set.

To demonstrate the impact of transfer learning, Table 1 presents a direct comparison between the performance of models trained from scratch on Polish data and those retrained using transfer learning from related Central European license plate formats (Czech, Hungarian, and Slovak). The results show consistent improvements across all evaluation measures after applying transfer learning, confirming its effectiveness in enhancing recognition accuracy and robustness.

It is worth noting that although the differences between the various transfer learning approaches are small, they result in a significant reduction in normed error, defined as 1-normED. For example, the original error for Polish license plates was 4.2%, which could be reduced by 28-38%.

These results suggest that transferring knowledge from closely related Central European plate formats (Czech, Hungarian, and Slovak) to Polish data can be beneficial, as all three retrained models converge to similar performance levels. Although the differences among them are relatively small, the Czech-based retraining yields marginally higher accuracy on the Polish validation set, while the Hungarian-based retraining achieves a slightly higher normED.

Table 2 provides a comparison between the presented results and those reported in the literature, using shared evaluation measures where possible.

As shown in Table 2, the proposed approach consistently outperforms most of the results reported in the literature for Polish license plate recognition. While several earlier studies focus primarily on image quality evaluation or system architecture without reporting concrete recognition measures – such as the works by Leszczuk et al. [28], Łubkowski et al. [52], and Jureczko et al. [19] – those that do provide quantitative performance data are included in this comparison.

Most of the studies utilise accuracy as the primary evaluation measure, typically under varying environmental conditions. This makes it possible to conduct a meaningful comparison with the results of the presented method. In this regard, the proposed approach outperforms the results of Kluwak et al. [25], Janowski et al. [18], and Wróblewski [46] in general scenarios. The only exception is Kluwak et al. [25], who report

Tab. 2. Overview of the proposed and literature results in Polish LPR.

Publication	**Measure**	**Result**	**Note**
Kluwak et al. [25]	Accuracy ITCC Accuracy	53.12%-84.10% 62.83%-90.96%	The accuracy ranges reflect variation across datasets under different environmental conditions. ITCC aggregates and corrects predictions across multiple frames, resulting in better performance.
Janowski et al. [18]	LANN CC accuracy PWPLD CC accuracy Human CC accuracy LANN CC $(1 + \text{LED})^{-1}$ PWPLD CC $(1+\text{LED})^{-1}$ LANN NC $(1 + \text{LED})^{-1}$ PWPLD NC $(1+\text{LED})^{-1}$	4.89% 1.67% 70.33% 63% 57% 93% 81%	Out of 900 sequences, the LANN and PWPLD methods correctly recognised only 44 and 15 cases, respectively, while human subjects achieved 633 correct responses. Due to the low accuracy of automatic methods, alternative measures were introduced to enable more nuanced performance comparison.
Kekez [21]	Character accuracy Character accuracy with correction	88.7% 99%	Based on number plate and character format correction rules, post-processing was implemented for the output of the recognition algorithm.
Wróblewski [46]	Accuracy	60%-70%	The accuracy range reflects variation in image quality.
Czech to Polish	Accuracy normED $(1 + \text{LED})^{-1}$ Character accuracy	89.2% 97.3% 96.8% 96.6%	Pretrained on Czech license plates and retrained on Polish ones.
Hungarian to Polish	Accuracy normED $(1 + \text{LED})^{-1}$ Character accuracy	88.0% 97.4% 97.0% 96.9%	Pretrained on Hungarian license plates and retrained on Polish ones.
Slovak to Polish	Accuracy normED $(1 + \text{LED})^{-1}$ Character accuracy	88.6% 97.0% 96.6% 96.5%	Pretrained on Slovak license plates and retrained on Polish ones.

higher accuracy for high-quality images under ideal lighting conditions by aggregating predictions across multiple frames using their ITCC technique. However, this specific scenario is addressed separately in the next subsection using a dedicated technique developed within this study. Overall, the presented solution demonstrates superior general performance.

In addition to accuracy, several publications also consider character-level measures. For instance, Janowski et al. [18] employ the $(1 + \text{LED})^{-1}$ measure, which is clearly outperformed by the proposed models. Kekez [21] reports a character accuracy of 88.7% under standard conditions, which is also exceeded by the presented approach. Although his post-processed version achieves up to 99% by applying character format correction

rules – such as measuring distances between specific points of characters – this relies in part on synthetic data, which reduces the relevance and generalizability of the result, especially in real-world scenarios.

Therefore, in more realistic application scenarios-where image quality, environmental conditions, and plate variability cannot be controlled-the presented approach emerges as the more robust and practically preferable solution.

4.3. Robustness to weather conditions

An additional experiment was conducted to evaluate the robustness of the proposed models in adverse weather and lighting conditions by synthetically modifying the Polish validation set using the Albumentations [6] library. Five image subsets were generated to simulate fog, rain, shadow, snow, and sunflare. Four models were tested on these subsets: one trained exclusively on Polish plates, and three produced by transferring knowledge from other Central European datasets – Czech, Hungarian, and Slovak – and subsequently retraining on Polish plates. Table 3 summarizes the accuracy obtained on these five subsets, while Table 4 reports the corresponding normED.

Overall, fog and rain tend to cause only moderate performance degradation compared to the original test conditions. For instance, when evaluating rain images, the transfer-based Czech to Polish and Hungarian to Polish models each achieve 76.51% accuracy, outperforming the purely Polish-trained model's 73.49%. The normED under rain likewise increases from 92.43% for the Polish to Polish baseline to around 93.3% for

Tab. 3. Accuracy under Adverse Conditions.

Subset	Continue of Polish (%)	Czech to Polish (%)	Hungarian to Polish (%)	Slovak to Polish (%)
Fog	73.49	76.51	72.29	71.69
Rain	73.49	76.51	76.51	76.51
Shadow	67.47	71.08	72.29	71.69
Snow	44.58	45.18	46.39	46.39
Sunflare	45.78	48.19	44.58	46.39

Tab. 4. NormED under Adverse Conditions.

Subset	Continue of Polish (%)	Czech to Polish (%)	Hungarian to Polish (%)	Slovak to Polish (%)
Fog	92.39	93.45	92.16	92.23
Rain	92.43	93.33	93.36	93.01
Shadow	91.57	92.89	92.92	93.01
Snow	73.67	75.24	73.91	73.74
Sunflare	70.74	72.01	70.92	71.25

Fig. 8. License Plates under Adverse Conditions.

both Czech to Polish and Hungarian to Polish, indicating more precise character-level predictions. Shadow similarly shows notable gains from transfer: its best-performing models attain 72.29% and 71.69% accuracy, respectively, compared to 67.47% for the Polish to Polish baseline.

A different pattern emerges in snow and sunflare conditions, where plate details can be heavily obscured. While accuracy remains relatively low overall, transfer still provides marginal improvements. Under snow, for example, the best-performing transfer-based model reaches 46.39% accuracy, whereas the baseline stays at 44.58%. In sunflare images, Czech to Polish yields 48.19%, exceeding the baseline of 45.78%. Nonetheless, it should be noted that not all transfer-trained models universally surpass the purely Polish model under every challenging scenario: for instance, while Czech to Polish outperforms Polish to Polish under fog, Hungarian to Polish falls slightly behind. Such variation underscores the complexity of weather-induced artifacts and suggests that future efforts could refine domain adaptation strategies or explore specialized attention mechanisms to further strengthen resilience when license plates are partially occluded, blurred, or distorted by inclement weather and harsh lighting. Figure 8 presents examples of each distortion category, illustrating how the applied simulations affect the visibility of license plates.

4.4. Testing on video sequences

To assess real-world applicability, the model that performed best on the Polish validation set-originally trained on Czech license plates and then retrained on Polish-was evaluated on 22 raw video sequences from the publicly available Parking Database [28]. Because these videos were recorded under natural conditions without preprocessing or image enhancements, they provide a more demanding test environment than the still-image validation dataset. As described earlier in Subsection 3.1, the final license plate prediction for each video was determined by applying a majority voting strategy across all frame-level predictions. This means that each frame was independently processed to

Tab. 5. Majority Votes.

Source (src)	Majority Vote	Frequency (%)	Second Vote	Frequency (%)	Third Vote	Frequency (%)
src1	kmi77ev	99.61	kmi77ee	0.39	–	–
src4	kr154gk	81.78	kr154ck	14.10	krise1	0.87
src5	kr439ha	75.72	kr439hal	19.93	kr439ma	4.35
src6	kr537cw	95.37	kr537ch	1.54	kr577ch	0.77
src7	kr539jk	64.77	kr5399k	20.35	kr5393k	2.84
src8	kr642lu	84.43	ar5411	0.94	vr642lu	0.71
src10	kr701kn	96.45	kr701kni	2.37	ka701kn	0.47
src11	kr731gw	93.79	vk731gw	0.59	pr771gm	0.30
src12	kr960mf	46.08	kry60mf	39.43	kr960mp	1.66
src13	kr992es	100.00	–	–	–	–
src15	kr2492k	100.00	–	–	–	–
src18	kr3527l	84.58	kr352yl	1.10	kavvl	0.66
src20	kr6986n	97.69	krr1114	0.29	kr65841	0.29
src21	kr8146m	92.88	kk8146m	2.67	wk8146m	1.78
src22	kr9195n	99.80	xr803lg	0.20	–	–
src23	kr9253g	98.15	kr9257g	1.23	wr9257g	0.21
src24	kr9611u	100.00	–	–	–	–
src25	kr9764s	50.51	kr97645	47.45	kr97649	0.41
src26	kra3hp2	81.13	krajhp2	13.73	krashp2	1.47
src27	kli05659	100.00	–	–	–	–
src28	kwia593	100.00	–	–	–	–
src30	w67045w	58.85	wig7045w	12.17	wug7045w	9.51
Average		86.40		8.10		1.20

yield a character string, and the most frequently occurring string in the entire sequence was selected as the final recognised plate.

A majority voting mechanism was used for each video source ID to stabilize predictions across frames, thereby reducing frame-by-frame recognition errors. In this context, the "Majority Vote" refers to the most frequently predicted license plate string across all frames of a given video, while the "Second Vote" and "Third Vote" denote the next most frequent predictions. Their corresponding frequencies indicate how dominant each prediction was relative to the total number of frames. Table 5 presents the top three majority votes for each source video along with their relative frequencies; the bottom row shows the overall accuracy of the majority-voting approach on all videos. So 22 videos were used in the testing phase and the identifiers of Table 5 follow the original coding scheme of the Parking Database [28], in which video sequences are labelled with non-sequential IDs such as src1, src4, src5, etc. This facilitates traceability and allows readers to map the presented results directly to the corresponding videos in the database.

Overall, the system achieves 86.4% accuracy on these raw test videos, which is about

1.8% lower than its performance on the Polish still-image validation set. This decline is attributed to real-world challenges such as variable lighting, motion blur, and occlusions that are not fully represented in the still-image dataset. Nonetheless, the majority voting method exhibits considerable robustness, demonstrating its ability to accurately identify plates under stable or moderately changing conditions. Moreover, for every tested sequence, the majority vote aligns with the ground truth plate, and thus – when only the ensemble output is considered – the overall accuracy reaches 100%. This means that, despite individual frame-level misclassifications—which are reflected in the presence of second and third votes – the most frequent prediction in each sequence always matches the ground truth.

While overall accuracy provides a broad measure of performance, it is critical to assess the system's behaviour in terms of specific error types, especially in real-world deployment scenarios where the consequences of false positives and false negatives differ substantially. In the context of automated access control, a false negative occurs when an authorised vehicle is incorrectly denied entry, whereas a false positive arises when an unauthorised vehicle is mistakenly admitted. Although both are technically misclassifications, their implications diverge – false negatives cause inconvenience, while false positives may result in security breaches.

To quantify these risks, we consider a typical operational setting in which a vehicle remains within the camera's field of view for approximately 2 seconds. This duration corresponds to $T = 50$ consecutive video frames at 25 frames per second. Each frame is processed independently, and the final decision is determined by a majority vote over all frame-level predictions. Given the observed average per-frame accuracy of $p = 0.864$, the number of correct predictions over the 50 frames can be modelled as a binomial random variable $X \sim \text{Binomial}(n = 50, p = 0.864)$. A false negative corresponds to the event in which fewer than 26 correct predictions are made, i.e., $\mathbb{P}(X < 26)$.

This demonstrates that the likelihood of rejecting an authorised vehicle, when aggregating predictions across 50 frames, is less than $8.4 \cdot 10^{-10}$, which is effectively zero in practice.

To estimate the false positive probability, we adopt a worst-case analysis. We assume that the system consistently misclassifies the observed unauthorised plate in every frame, and furthermore, that all misclassifications result in the same incorrect prediction—one that coincidentally belongs to the authorised set. This corresponds to a per-frame probability of consistent misclassification equal to $p = 1 - 0.864 = 0.136$. The number of times this incorrect (but authorised) plate appears as a prediction over 50 frames is modelled as $Y \sim \text{Binomial}(n = 50, p = 0.136)$, and a false positive occurs if at least 26 such misclassifications arise, i.e., $\mathbb{P}(Y \geq 26)$.

This demonstrates that the likelihood of accepting an unauthorised vehicle, when aggregating predictions across 50 frames, is less than $1.3 \cdot 10^{-10}$, which is effectively zero in practice as well.

These results indicate that, even under highly pessimistic assumptions, the probability of both rejecting an authorised plate and accepting an unauthorised one remains vanishingly small. The use of majority voting across a short temporal window provides not only improved robustness but also strong statistical guarantees against both types of critical classification errors in security-sensitive applications.

5. Conclusion and future work

This study presents a novel CNN-RNN-based framework tailored to license plate recognition in Poland, addressing the absence of recurrent neural network approaches in contemporary Polish LPR literature. By combining a Thin-Plate Spline module, a ResNet feature extractor, a BiLSTM sequence model, and an attention-based decoder, the proposed method capitalizes on sequence labelling to mitigate the segmentation bottlenecks common to traditional optical character recognition workflows. Comparative evaluations across Czech, Hungarian, Polish, and Slovak datasets underscore the model's robust performance, with transfer learning revealing notable improvements when retraining non-Polish models on Polish data. Moreover, testing on real-world parking-lot video sequences validates the adaptability of the proposed solution to dynamic conditions, reinforcing the benefit of majority voting to enhance recognition stability. Ultimately, the findings illustrate that the synergy of CNN-RNN architectures and transfer learning serves as an effective pathway for advancing the performance and resilience of Polish LPR systems.

Future work may also examine a broader spectrum of real-world applications to underscore the significance of rapid and precise license plate recognition. As an increasing number of cities implement environmental zones to mitigate air pollution and traffic congestion, accurate LPR systems can play a critical role in regulating and enforcing vehicle entry based on emissions standards. This functionality will likely become more pertinent as such zoning regulations proliferate in additional regions, including Poland. Beyond environmental monitoring, further applications include integration into advanced parking systems, where the technology could facilitate real-time detection of available spaces, guide drivers accordingly, and streamline the administration of car sharing or rental fleets. In the logistics and supply chain domains, improved plate recognition can enhance fleet tracking and management, while in residential and commercial contexts, the technology may be incorporated into automated gates or smart-home platforms to enable secure, seamless access for approved vehicles. Law enforcement, public safety, and public transportation authorities may likewise benefit from improved accuracy and robustness across large-scale, real-time deployments under diverse environmental conditions. By pursuing domain-specific data augmentation strategies, examining more sophisticated attention mechanisms, and conducting expanded empirical investigations,

future research can aim to address the persistent challenges of plate variability and distortion – ultimately fostering faster and more reliable LPR solutions for an increasingly diverse range of use cases.

Acknowledgement

This research was supported by the Wacław Felczak Foundation through the Jagelló grant program, designed to assist doctoral students conducting research related to Poland at Hungarian higher education institutions.

Project no. KDP-IKT-2023-900-I1-00000957/0000003 has been implemented with the support provided by the Ministry of Culture and Innovation of Hungary from the National Research, Development and Innovation Fund, financed under the Co-operative Doctoral Program [C2269690] funding scheme.

References

[1] K. Ahmad, H. Khujamatov, A. Lazarev, N. Usmanova, M. Alduailij, et al. Internet of Things-Aided Intelligent Transport Systems in Smart Cities: Challenges, Opportunities, and Future. *Wireless Communications and Mobile Computing* 2023(1):7989079, 2023. doi:10.1155/2023/7989079.

[2] M. Arsenović, S. Sladojević, A. Anderla, and D. Stefanović. Deep learning driven plates recognition system. In: *17th International Scientific Conference on Industrial Systems*, pp. 1–4, 2017. https://www.iim.ftn.uns.ac.rs/is17/papers/29.pdf.

[3] J. Baek, G. Kim, J. Lee, S. Park, D. Han, et al. What is wrong with scene text recognition model comparisons? dataset and model analysis. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4714–4722, 2019. doi:10.1109/ICCV.2019.00481.

[4] G. Bradski. The openCV library. *Dr. Dobb's Journal: Software Tools for the Professional Programmer* 25(11):120–123, 2000.

[5] A. C. Bukola, P. A. Owolawi, C. Du, and E. Van Wyk. A systematic review and comparative analysis approach to boom gate access using plate number recognition. *Computers* 13(11):286, 2024. doi:10.3390/computers13110286.

[6] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, et al. Albumentations: fast and flexible image augmentations. *Information* 11(2):125, 2020. doi:10.3390/info11020125.

[7] N. Cahyadi, S. Salsabila, S. Raniprima, and V. Monita. Review of license plate recognition techniques with deep learning. *Jurnal Teknologika* 14(2):458–469, 2024. https://jurnal.wastukancana.ac.id/index.php/teknologika/article/view/415.

[8] L. T. A. Dang, V. D. Ngoc, P. C. L. Thien Vu, N. N. Truong, and T. D. Trinh. Vietnam vehicle number recognition based on an improved CRNN with attention mechanism. *International Journal of Intelligent Transportation Systems Research* pp. 1–16, 2024. doi:10.1007/s13177-024-00402-7.

[9] P. Das, S. Mitra, S. Chakraborty, M. H. K. Mehedi, M. Y. M. Adib, et al. CNN-GRU based fusion architecture for bengali license plate recognition with explainable AI. In: *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pp. 1–6, 2023. doi:10.1109/ICCCNT56998.2023.10413582.

[10] European Commission. Road freight transport by type of goods – annual data. Eurostat, 2024. https://ec.europa.eu/eurostat/databrowser/view/road_go_ta_dc/. [Accessed: Jan 1, 2025].

[11] A. Graves and J. Schmidhuber. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks* 18(5-6):602–610, 2005. doi:10.1016/j.neunet.2005.06.042.

[12] J. Gąsienica-Józkowy, M. Knapik, and B. Cyganek. An ensemble deep learning method with optimized weights for drone-based water rescue and surveillance. *Integrated Computer-Aided Engineering* 28(3):221–235, 2021. doi:10.3233/ICA-210649.

[13] S. Hadavi, H. B. Rai, S. Verlinde, H. Huang, C. Macharis, et al. Analyzing passenger and freight vehicle movements from automatic-number plate recognition camera data. *European Transport Research Review* 12:1–17, 2020. doi:10.1186/s12544-020-00405-x.

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. doi:10.1109/CVPR.2016.90.

[15] K. Hefnawy, A. Lila, E. Hemayed, and M. Elshenawy. A robust license plate detection and recognition framework for arabic plates with severe tilt angles. *International Journal of Advanced Computer Science & Applications* 15(2), 2024. doi:10.14569/IJACSA.2024.0150287.

[16] C. Y. Huang. A review of deep learning for character recognition area. In: *The 26th International Scientific and Practical Conference "World Problems and Ways of Solving Modern Problems"*, p. 245, 2024. doi:10.46299/ISG.2024.1.26.

[17] Jaided AI. Easyocr (version 1.6.2) [computer software], 2020. https://github.com/JaidedAI/EasyOCR. [Accessed: May 27, 2025].

[18] L. Janowski, P. Kozłowski, R. Baran, P. Romaniak, A. Glowacz, et al. Quality assessment for a visual and automatic license plate recognition. *Multimedia Tools and Applications* 68:23–40, 2014. doi:10.1007/s11042-012-1199-5.

[19] M. Jureczko and F. Uherek. The automatic license plate recognition system based on video sequences using artificial neural networks. In: *Modelling in Engineering 2020: Applied Mechanics*, pp. 95–106, 2021. doi:10.1007/978-3-030-68455-6_9.

[20] F. Kahraman, B. Kurt, and M. Gökmen. License plate character segmentation based on the Gabor transform and vector quantization. In: *International Symposium on Computer and Information Sciences*, pp. 381–388, 2003. doi:10.1007/978-3-540-39737-3_48.

[21] M. Kekez. An approach to license plate recognition in real time using multi-stage computational intelligence classifier. *International Journal of Electronics and Telecommunications* pp. 275–280, 2023. doi:10.24425/ijet.2023.144361.

[22] M. M. Khan, M. U. Ilyas, I. R. Khan, S. M. Alshomrani, and S. Rahardja. License plate recognition methods employing neural networks. *IEEE Access* 11:73613–73646, 2023. doi:10.1109/ACCESS.2023.3254365.

[23] Z. H. Khattak, B. L. Smith, M. D. Fontaine, J. Ma, and A. J. Khattak. Active lane management and control using connected and automated vehicles in a mixed traffic environment. *Transportation Research Part C: Emerging Technologies* 139:103648, 2022. doi:10.1016/j.trc.2022.103648.

[24] W. Kim and G. L. Chang. Development of a hybrid prediction model for freeway incident duration: A case study in maryland. *International Journal of Intelligent Transportation Systems Research* 10:22–33, 2012. doi:10.1007/s13177-011-0039-8.

[25] K. Kluwak, J. Segen, M. Kulbacki, A. Drabik, and K. Wojciechowski. ALPR-extension to traditional plate recognition methods. In: *Intelligent Information and Database Systems: 8th Asian Conference (ACIIDS 2016)*, pp. 755–764, 2016. doi:10.1007/978-3-662-49390-8_73.

[26] E. Kot, Z. Krawczyk, K. Siwek, L. Królicki, and P. Czwarnowski. Deep learning-based framework for tumour detection and semantic segmentation. *Bulletin of the Polish Academy of Sciences Technical Sciences* p. e136750, 2021. doi:10.24425/bpasts.2021.136750.

[27] R. Laroca, E. Severo, L. A. Zanlorensi, L. S. Oliveira, G. R. Gonçalves, et al. A robust real-time automatic license plate recognition based on the YOLO detector. In: *2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–10, 2018. doi:10.1109/IJCNN.2018.8489629.

[28] M. Leszczuk, L. Janowski, J. Nawała, J. Zhu, Y. Wang, et al. Objective video quality assessment and ground truth coordinates for automatic license plate recognition. *Electronics* 12(23):4721, 2023. doi:10.3390/electronics12234721.

[29] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. *Doklady Akademii Nauk* 163(4):845–848, 1965. http://mi.mathnet.ru/dan31411. [Accessed: Apr 12, 2025].

[30] D. Li and J. Lasenby. Mitigating urban motorway congestion and emissions via active traffic management. *Research in Transportation Business & Management* 48:100789, 2023. doi:10.1016/j.rtbm.2022.100789.

[31] H. Li, P. Wang, and C. Shen. Toward end-to-end car license plate detection and recognition with deep neural networks. *IEEE Transactions on Intelligent Transportation Systems* 20(3):1126–1136, 2018. doi:10.1109/TITS.2018.2847291.

[32] T. Lindner, D. Wyrwał, M. Białek, and P. Nowak. Face recognition system based on a single-board computer. In: *2020 International Conference Mechatronic Systems and Materials (MSM)*, pp. 1–6, 2020. doi:10.1109/MSM49833.2020.9201668.

[33] S. Z. Masood, G. Shu, A. Dehghan, and E. G. Ortiz. License plate detection and recognition using deeply learned convolutional neural networks. arXiv, arXiv:1703.07330, 2017. doi:10.48550/arXiv.1703.07330.

[34] Minister Infrastruktury. Rozporządzenie ministra infrastruktury z dnia 8 listopada 2024 r. w sprawie rejestracji i oznaczania pojazdów, wymagań dla tablic rejestracyjnych oraz wzorów innych dokumentów związanych z rejestracją pojazdów. Dz.U. 2024 poz. 1709, 2024. https://isap.sejm.gov.pl/isap.nsf/DocDetails.xsp?id=WDU20240001709. (Minster of Infrastructure. Regulation of the Minister of Infrastructure of 8 November 2024 on the registration and marking of vehicles, requirements for license plates and the use of other documents related to vehicle registration).

[35] A. Mousa. Canny edge-detection based vehicle plate recognition. *International Journal of Signal Processing, Image Processing and Pattern Recognition* 5(3):1–8, 2012. https://staff.najah.edu/media/sites/default/files/Canny_Edge-Detection_Based_Vehicle_Plate_Recognition.pdf.

[36] D. Moussa, A. Maier, F. Schirrmacher, and C. Riess. Sequence-based recognition of license plates with severe out-of-distribution degradations. In: *International Conference on Computer Analysis of Images and Patterns*, pp. 175–185, 2021. doi:10.1007/978-3-030-89131-2_16.

[37] V. Naosekpam, A. S. Shishir, and N. Sahu. Scene text recognition with orientation rectification via IC-STN. In: *TENCON 2021 – IEEE Region 10 Conference (TENCON)*, pp. 664–669, 2021. doi:10.1109/TENCON54134.2021.9707183.

[38] OpenALPR. Automatic license plate recognition. Rekor, 2015. http://www.openalpr.com/. [Accessed: Apr 12, 2025].

[39] V. H. Pham, P. Dinh, and V. H. Nguyen. CNN-based character recognition for license plate recognition system. In: *Asian Conference on Intelligent Information and Database Systems*, 2018. doi:10.1007/978-3-319-75420-8_56.

[40] P. S. Puri and P. P. Bartakke. SSD based vehicle number plate detection and recognition. In: *2023 International Conference on Network, Multimedia and Information Technology (NMITCON)*, pp. 1–6, 2023. doi:10.1109/NMITCON58196.2023.10276329.

[41] See5. Data mining tool. Rulequest Research, 2012. http://www.rulequest.com/products.html. [Accessed: Apr 12, 2025].

[42] R. Smith. An overview of the Tesseract OCR engine. In: *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, pp. 629–633, 2007. doi:10.1109/ICDAR.2007.4376991.

[43] S. Srinivasan, D. Prabha, N. M. Raffic, K. G. Babu, S. Thirumurugaveerakumar, et al. Automated vehicle number plate recognition system using convolution long short-term memory technique. In: *Object Detection with Deep Learning Models*, pp. 101–115. Chapman and Hall/CRC, 2022. doi:10.1201/9781003206736-6.

[44] Y. Wang, X. Yang, H. Liang, and Y. Liu. A review of the self-adaptive traffic signal control system based on future traffic environment. *Journal of Advanced Transportation* 2018(1):1096123, 2018. doi:10.1155/2018/1096123.

[45] A. Wilkowski, M. Stefańczyk, and W. Kasprzak. Training data extraction and object detection in surveillance scenario. *Sensors* 20(9):2689, 2020. doi:10.3390/s20092689.

[46] P. Wróblewski. Recognition of polish car license plates. In: *Computer Recognition Systems: Proceedings of the 4th International Conference on Computer Recognition Systems (CORES'05)*, pp. 893–900, 2005. doi:10.1007/3-540-32390-2_105.

[47] F. You, Y. Zhao, and X. Wang. Combination of CNN with GRU for plate recognition. *Journal of Physics: Conference Series* 1187(3):032008, 2019. doi:10.1088/1742-6596/1187/3/032008.

[48] M. D. Zeiler. ADADELTA: An adaptive learning rate method. arXiv, arXiv:1212.5701, 2012. doi:10.48550/arXiv.1212.5701.

[49] H. Zhao, C. Song, H. Zhao, and S. Zhang. License plate recognition system based on morphology and LS-SVM. In: *2008 IEEE International Conference on Granular Computing*, pp. 826–829, 2008. doi:10.1109/GRC.2008.4664738.

[50] J. Zhao, W. Li, J. Wang, and X. Ban. Dynamic traffic signal timing optimization strategy incorporating various vehicle fuel consumption characteristics. *IEEE Transactions on Vehicular Technology* 65(6):3874–3887, 2015. doi:10.1109/TVT.2015.2506629.

[51] L. Zou, Z. He, K. Wang, Z. Wu, Y. Wang, et al. Text recognition model based on multi-scale fusion CRNN. *Sensors* 23(16):7034, 2023. doi:10.3390/s23167034.

[52] P. Łubkowski and D. Laskowski. Assessment of quality of identification of data in systems of automatic licence plate recognition. In: *Smart Solutions in Today's Transport: 17th International Conference on Transport Systems Telematics (TST 2017)*, pp. 482–493, 2017. doi:10.1007/978-3-319-66251-0_39.