Vol. 35, No. 1, 2026

# 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

# Enhancing Cultural Heritage Digitalization through 3D Graphics Algorithm and Immersive Visual Communication Technology

Fang Yuan*
*Guangxi Normal University for Nationalities, College of Art, Chongzuo, China*
**Corresponding author: Fang Yuan (yuanfang16316@163.com)*

**Submitted:** 17 Jun 2025 **Accepted:** 06 Oct 2025 **Published:** 16 Feb 2026 

**Abstract** With the continuous advancement of digital technology, cultural and creative product design is shifting from static presentation to dynamic immersive experience. The research aims to address the challenges faced by traditional modeling methods in accurately restoring complex textures and cross platform visual communication. The neural radiation field algorithm was enhanced by introducing a multi-level cost volume fusion module and a Gaussian uniform mixture sampling strategy. Furthermore, a collaborative visual communication framework integrating augmented reality and virtual reality was constructed, achieving a transition from single image input to high-precision 3D reconstruction, and then to dynamic interaction. The experiment showed that the improved algorithm achieved peak signal-to-noise ratios of 30.63 and 30.15 on the UoM-Culture3D and Bootstrap 3D synthetic datasets, respectively, with structural similarity indices of 0.88 and 0.89, respectively. Field deployment tests have shown that integrating AR and VR technologies into visual communication strategies significantly improves spatial perception consistency, prolongs user engagement time, and enhances detail recognition accuracy. This research emphasizes the potential of combining deeply coupled 3D graphics algorithms with immersive technology, which can help improve the digital restoration accuracy and cultural dissemination efficiency of cultural and creative products, thereby supporting the modern inheritance of traditional culture.

**Keywords:** 3D graphics algorithm, visual communication technology, cultural and creative product design, NeRF, VR, AR.

## 1. Introduction

With the adoption of digital technology in the industry, cultural and creative product design is facing a transition from static output to dynamic immersive experience. The popularization of Virtual Reality (VR) hardware and the advancement of real-time graphics computing have made the digital revitalization of cultural heritage a new direction. Through technological means, it is possible to break through the physical limitations of physical exhibitions, allowing historical patterns and traditional techniques to gain cross temporal and spatial dissemination power. This cultural and creative product design has put forward new requirements and urgently needs to break through the limitations of traditional two-dimensional expression, establish a multidimensional design system that integrates high-precision modeling, dynamic narrative, and interactive experience [14]. Currently, the Neural Radiation Field (NeRF) technology in the field of 3D graphics algorithms combines ray tracing and deep learning to achieve high fidelity digital reconstruction of complex cultural carriers such as cultural relics patterns and historical

scenes [21]. However, this technology relies on dense input of hundreds of images in a single scene and time-consuming training on a scene by scene basis, making it difficult to adapt to the fast iterative design process of cultural and creative products [12, 31]. The augmented reality (AR) and VR technologies in the field of visual communication can create a virtual real fusion experience environment. However, most of the existing schemes use a single mode, which has problems such as large spatial alignment error, homogenization of interaction forms, and shallow semantic analysis of cultural symbols [27, 32]. To this end, a multidimensional design method for cultural and creative products based on the Improved NeRF (INeRF) algorithm and the integration of AR and VR is proposed. By integrating multi-level cost structures and utilizing cross-scale feature fusion techniques, geometric reasoning capabilities are strengthened. Furthermore, the implementation of a Gaussian uniform mixture sampling strategy optimizes the efficiency of surface detail reconstruction. Consequently, a seamless interactive experience across AR and VR platforms is attained within the visual communication layer. The research aims to enhance the cultural connotation expression and user experience of cultural and creative products, and promote the development of the cultural and creative industry towards digitalization and multidimensionality. The innovation of the research lies in introducing a multi-level geometric feature fusion mechanism and a mixed sampling strategy into the NeRF framework. Meanwhile, through AR-VR collaborative interactive design, the organic unity of cultural symbols in spatial, temporal, and perceptual dimensions is achieved, providing practical and expressive methodological support for the digital innovation of cultural and creative products.

## 2. Related works

High-precision 3D reconstruction is the cornerstone of cultural heritage digitization. NeRF technology has garnered significant attention for its ability to fuse ray tracing with deep learning, enabling high-fidelity reconstruction of the complex textures and structures of cultural relics. However, classical NeRF and its variants generally suffer from significant limitations: their training process heavily relies on hundreds of dense multi-view images from a single scene and time-consuming scene-by-scene optimization, which severely restricts their applicability in cultural and creative product design workflows requiring rapid iteration. To address reconstruction challenges in specific domains, researchers have proposed various optimization schemes. To achieve texture synthesis optimization, Houdard et al. [9] proposed a general framework named GOTEX. By constraining the local feature statistical distribution and utilizing the optimal transport semi-dual formula to control the feature distribution, high-quality texture synthesis and restoration were achieved. To improve the accuracy and efficiency of 3D reconstruction of ancient buildings, Ge et al. [7] introduced depth supervision into the NeRF framework,

combining a truncated signed distance function and an incremental training strategy, effectively enhancing the accuracy and efficiency of 3D reconstruction of ancient buildings. In the field of dynamic scene reconstruction, Qiu et al. [19] innovatively combined NeRF with signed distance fields to achieve realistic reconstruction of dynamic ship models, demonstrating its potential for dynamic modeling of specific objects. Mazzacca et al. [15] further validated the effectiveness of NeRF in reconstructing cultural heritage datasets, particularly in handling uniform textures or shiny surfaces, expanding the documentation pathways for digital heritage.

Visual communication technology serves as a bridge connecting digital reconstruction outcomes with user experience. AR and VR technologies enable the creation of immersive cultural experience environments that blend virtual and real elements. To enhance the visual communication effectiveness of digital animated advertisements, Fang et al. [5] proposed a multimodal visual communication system model based on multimodal video emotion analysis. This model dynamically adjusts digital animated advertisement content according to user emotions, enhancing the personalization and appeal of interactions, and demonstrating the potential of emotion-driven content adaptation. Liu et al. [13] conducted an in-depth analysis of visual communication strategies for cultural imagery in rural environments, emphasizing the importance of environmental perception in experiencing cultural spirit through the integration of art intervention institutions, and providing insights for cultural narratives in specific spaces. In terms of communication effectiveness evaluation, the video data analysis system by Yachnaya et al. [26] can identify and assess paralinguistic and non-verbal components in communication, providing tools for quantifying user experience. Yudhanto et al. [30] advocate a visual communication design philosophy grounded in culture and communication, emphasizing the importance of researching the target audience's values, norms, language, beliefs, and visual elements to enhance the cultural relevance and effectiveness of design.

As can be seen from the above, although three-dimensional graphics algorithms and visual communication technologies have made significant progress in their respective fields, there remains a lack of cross-platform, multi-modal integrated design methods for the digitization of cultural heritage. Existing solutions often struggle to balance high-precision texture restoration, real-time interactive performance, and visual consistency across multiple devices. This research gap leads to issues such as experience discontinuity and information loss in the actual dissemination of cultural and creative products. To address this, the study proposes a multi-dimensional design framework for cultural and creative products based on an improved INeRF and the deep integration of AR and VR, providing a solution that combines precision and expressiveness for the innovative transformation and dissemination of cultural heritage.

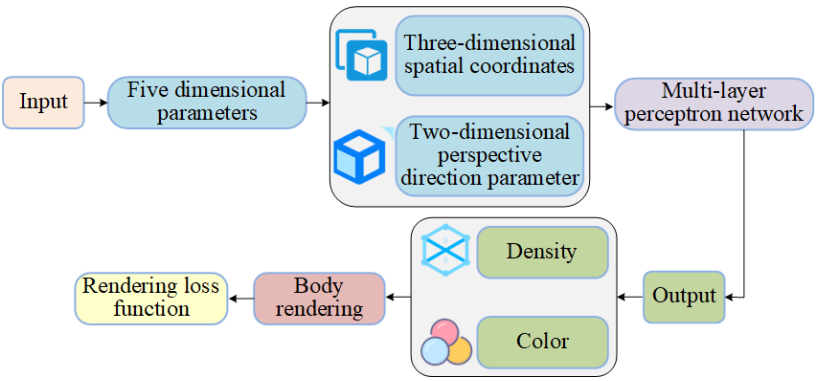


Fig. 1. Schematic diagram of NeRF algorithm (icons designed by Freepik [6]).

## 3. Methods and materials

### 3.1. Design of 3D graphics algorithm based on INeRF

Three-dimensional graphics algorithms are driving the transformation of cultural and creative products toward multi-dimensional design. NeRF technology combines deep learning and ray tracing to achieve high-fidelity three-dimensional reconstruction of cultural relics and historical scenes, effectively restoring complex textures and material effects, and solving the challenges of traditional modeling in reproducing complex materials and intricate patterns [11, 16, 28]. The basic structure of NeRF technology is shown in Fig. 1. This technology first receives a five-dimensional input parameter consisting of spatial position coordinates and the angle of light incidence. This parameter is then mapped by a multi-layer perceptron network into RGB color values and density parameters. Subsequently, the system emits rays from the viewpoint, continuously sampling points along the path, and uses a volume rendering formula to calculate the transmittance and color contribution of each point, thereby synthesizing a realistic lighting effect. Finally, the model is optimized using a pixel-level rendering loss function to approximate the optical properties of the real-world scene. Among them, the NeRF mapping function [10] is

$$F(x, y, z, \theta, \phi) \rightarrow (R, G, B, \sigma)\,, \tag{1}$$

where $x$, $y$, and $z$ represent three-dimensional spatial coordinates, $\theta$ and $\phi$ represent the angle parameters of the incident direction of light rays, $R$, $G$, and $B$ represent the RGB color values of the sampling points, and $\sigma$ represents the medium density of the sampling point. The function predicts the optical properties of each sampling point based on the light and scene geometry characteristics, thereby providing basic data for

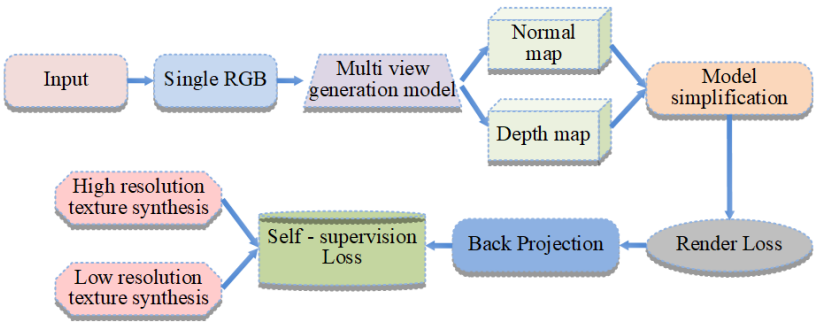


Fig. 2. The basic framework of INeRF.

volume rendering. The rendering expression is

$$C(r) = \int_{t_n}^{t_f} T(t) \cdot \sigma(r(t)) \cdot c(r(t), d)\, \mathrm{d}t\,, \tag{2}$$

where $C(r)$ represents the cumulative color of light, $T(t)$ indicates the transmittance of light from the starting point to the current point, $\sigma(r(t))$ represents the density of path point $r(t)$, $c(r(t), d)$ indicates the color of the path point $r(t)$ in the direction $d$, and $t_n$ and $t_f$ represent the starting and ending points of the light. This formula achieves optically realistic image synthesis by accumulating the color and transparency of each sampling point along the light path. The expression of the rendering loss function is

$$\mathcal{L}_{\text{render}} = \sum_{p} \left\| \hat{C}(p) - C_{\text{gt}}(p) \right\|^2 , \tag{3}$$

where $\mathcal{L}_{\text{render}}$ represents pixel-level rendering loss, $\hat{C}(p)$ represents the color of pixels in the generated image, and $C_{\text{gt}}(p)$ represents the color of pixel $p$ in real multi-view images. Although NeRF technology can achieve high-precision 3D reconstruction, it relies on a large number of input images from a single scene and time-consuming scene by scene optimization training, which makes it difficult to meet the design requirements for rapid iteration of cultural and creative products. Therefore, the study proposed the INeRF algorithm, whose basic framework is shown in Figure 2.

The INeRF algorithm starts with a single RGB input and extends the model to multi view data through multi view generation. It combines camera parameters to drive the 3D reconstruction module to generate normal maps and depth maps. During the process, supervised and soft supervised loss optimization is used to optimize depth and RGB prediction, and geometric consistency is ensured through backprojection. The rendering loss function further optimizes the lighting and material performance of the

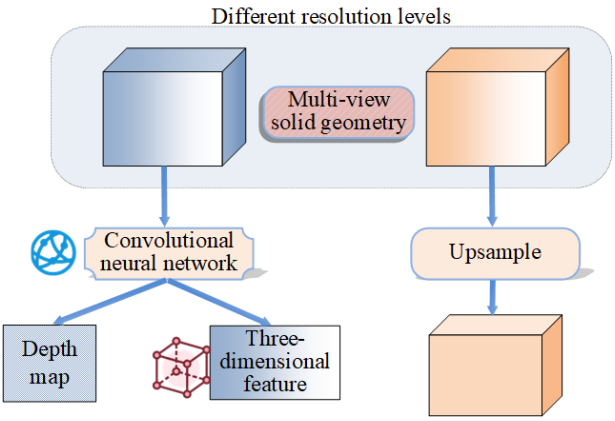


Fig. 3. The basic structure of the cost body (icons designed by Freepik [6]).

model, followed by high-resolution texture synthesis to enhance details, and ultimately balances accuracy and efficiency through model simplification techniques to output high-quality 3D models.

To address the issue of insufficient geometric information in single-view input, a multi-level cost volume fusion module based on convolutional attention was designed, as shown in Figure 3. Its the schematic diagram is based on multi-view solid geometry. Firstly, feature maps are extracted from input images of different resolution levels, and the three-dimensional geometric information of the scene is captured by constructing a multi-scale cost volume. In the feature fusion stage, low-resolution cost bodies encode global semantics, while high-resolution cost bodies retain details. Cross-layer feature interaction is achieved through convolutional attention, and channel and spatial attention are used to optimize the coordination of local and global information. Ultimately, a geometric neural field with both spatial accuracy and semantic integrity is formed, providing multi-level feature support for rendering. The formula for multi-level cost volume fusion is

$$F_{\text{fused}} = \sum_{l=1}^{L} w_l \cdot F_l^{\text{up}} + F_{\text{res}} \,, \tag{4}$$

where $F_{\text{fused}}$ represents the fused multi-level features, $F_l^{\text{up}}$ represents the features after upsampling at layer $l$, $w_l$ represents the feature weight calculated through attention mechanisms, $F_{\text{res}}$ represents the residual connection feature, and $L$ indicates the total number of feature levels. In the feature decoding and rendering optimization stage, IN-eRF achieves efficient and accurate volume rendering by improving the sampling strategy and loss function design. In response to the problem of insufficient density in traditional

uniform sampling, a Gaussian uniform mixture sampling strategy is proposed. Based on the depth prior information inferred from multi-view solid geometry, Gaussian distribution dense sampling is used in the surface area of the object, while maintaining uniform sampling density in non critical areas. The expression for Gaussian uniform mixture sampling distribution is [18]

$$P(s) = \lambda \cdot \mathcal{N}(s \mid \mu_d, \sigma_d) + (1 - \lambda) \cdot \mathcal{U}(s \mid s_{\min}, s_{\max}), \tag{5}$$

where $P(s)$ represents the probability density function of the sampling point $s$, $\mathcal{N}$ is the Gaussian distribution, $\mathcal{U}$ represents the uniform distribution, $\lambda$ represents mixed weight coefficients, $\mu_d$ represents the depth mean, and $\sigma_d$ represents variance.

Meanwhile, a deep self-supervised loss function was designed to generate pseudo depth maps using multi-view consistency constraints. The pixel information of the source view was distorted to the target perspective through differentiable reprojection, and a self-supervised signal without the need for real depth annotation was constructed. Moreover, during the feature decoding stage, the algorithm spatially aligns the three-dimensional local features generated by the geometric neural field with the two-dimensional global features. It then incorporates the encoded information of light ray directions, dynamically decoding the color and density values for each sampling point via a multi-layer perceptron. Finally, it synthesizes the pixel color and depth information of the target viewpoint using a differentiable rendering equation, thereby establishing an end-to-end trainable framework. Through this framework, designers can quickly convert historical images, physical photos, or 2D drawings into interactive 3D models, greatly improving the responsiveness and flexibility of the creative production process.

The pseudocode of the INeRF algorithm is presented in the Appendix A.

### 3.2. Design of cultural and creative products based on visual communication technology

After completing high-precision digital reconstruction based on 3D graphics algorithms, visual communication technology has become the core supporting means in multi-dimensional expression of cultural and creative products. To achieve deep dissemination and innovative expression of cultural values, a deep integration strategy based on AR and VR has been studied and designed. The overall framework is shown in Figure 4. In the data generation layer, the system relies on the INeRF algorithm to construct a high-precision 3D model from a single image, obtaining multidimensional data including geometry, normal maps, and depth maps, laying the foundation for subsequent visual presentation. The visual expression layer focuses on the graphic rendering and semantic visualization processing of 3D models, mapping digital models into recognizable and culturally significant visual content through lighting simulation, material mapping, and color coding, and adapting to AR and VR platforms for dynamic presentation [29].

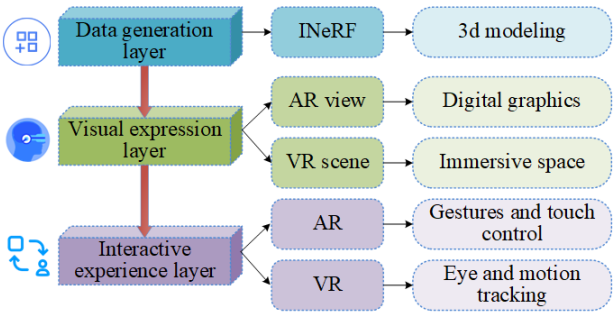


Fig. 4. The overall framework of visual communication strategy of cultural and creative products (icons designed by Freepik [6]).

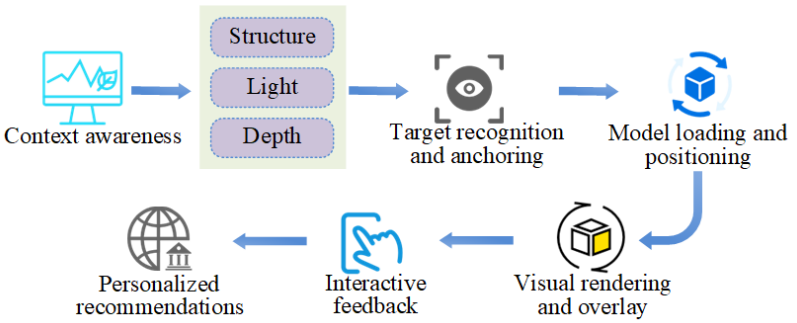


Fig. 5. Visual communication flow chart of AR-based cultural and creative products (icons designed by Freepik [6]).

The interactive experience layer revolves around user perception, combining the real-time positioning and virtual real overlay capabilities of AR, as well as the immersive spatial construction characteristics of VR, to achieve dynamic calling and multi-modal interaction design of cultural and creative graphic content.

The basic process of visual communication for AR-based cultural and creative products is shown in Figure 5. The system uses the RGB-D sensor built into the AR device to collect data on the geometric structure, depth distribution, and lighting conditions of the user's surroundings. It then uses feature point matching algorithms to identify and anchor targets, accurately locating physical objects such as display cases, cultural and creative packaging, and interior walls, and setting attachment points for virtual elements [22]. During the graphic deployment phase, the 3D models generated by INeRF are compressed and optimized for lightweight performance, then loaded into the augmented reality platform. The system automatically adjusts the orientation based on the on-site

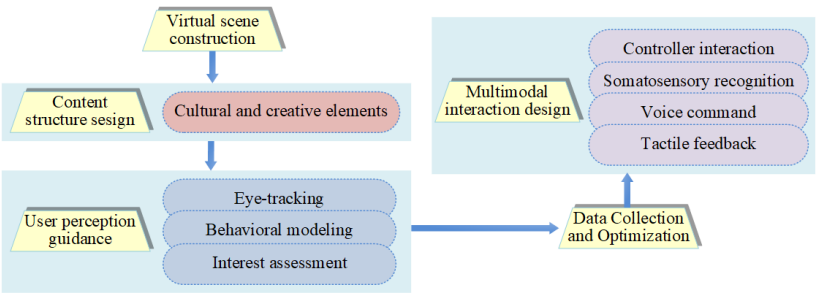


Fig. 6. Framework diagram of cultural and creative space construction based on VR (icons designed by Freepik [6]).

coordinate system. Subsequently, the system performs real-time graphic rendering and visual overlay, utilizing dynamic lighting estimation and reflection maps to ensure high consistency between virtual images and the real-world environment. Users can interact with virtual graphics through gesture recognition, voice input, or touch operations to obtain multi-dimensional feedback. The system finally combines user behavior trajectories and preference patterns to achieve personalized push notifications for cultural and creative content, further enhancing the targeting and engagement of visual communication [4]. To achieve seamless integration between virtual and real environments, the study developed an AR-VR hybrid interaction framework. When users transition from an AR scene to a VR scene, the system retains their operational state and interaction history through a spatial state caching mechanism, enabling state restoration and content continuity within the virtual space. First, the system uses the RGB-D sensor and IMU data from AR devices for real-time environmental mapping and user localization. Second, a virtual scene mapping model is established on the VR platform to ensure that the scene geometry aligns with the real-world spatial coordinates [3]. Finally, a state caching and synchronization mechanism is designed to save user interaction operations and object states, enabling seamless cross-device switching. In terms of on-site deployment, the system considers lighting matching, dynamic occlusion handling, and device load optimization to ensure stable operation in exhibition or cultural and creative experience spaces.

The AR-VR hybrid interaction framework is shown in Figure 6, which is the framework for constructing cultural and creative spaces based on VR. It systematically outlines the methodological path of VR technology in multidimensional cultural and creative design. Firstly, the designer relies on a 3D model database and INeRF generated results to construct a virtual environment that covers historical block restoration, cultural festival scenes, and immersive exhibition spaces for cultural relics, forming a virtual field with cultural depth. At the level of content structure, cultural and creative elements are

orderly embedded into spatial nodes, forming multiple types of information units, including decorative shapes, interactive objects, semantic labels, and dynamic animations, thus establishing a rich cultural narrative space. The system integrates gaze tracking and behavior modeling modules to dynamically adjust the visual hierarchy and dynamic parameters of virtual content based on users' attention paths and interest preferences, guiding users to naturally integrate into the narrative process. In terms of interaction, the platform integrates controller control, speech recognition, motion capture, and tactile feedback technology to provide users with multi-channel immersive interaction methods, enhancing the degree of freedom and realism of the experience. Meanwhile, the system continuously collects user behavior data in the virtual space in the background, including field of view movement, dwell time, and interaction frequency, providing data support and model basis for subsequent scene structure adjustment and visual information optimization, thereby achieving iterative updates and precise push of the design system.

## 4. Results

### 4.1. Performance verification of 3D graphics algorithm based on INeRF

To verify the effectiveness of multi-dimensional design of cultural and creative products based on 3D graphics algorithms and visual communication technology, a 3D reconstruction and visual communication system for cultural and creative products based on INeRF algorithm and AR/VR fusion was constructed in an experimental environment with GPU acceleration capability.

The image datasets used in the experiment included the UoM-Culture3D dataset [25] and the Bootstrap3D synthetic dataset [23, 24]. The UoM-Culture3D dataset contains multi-perspective images of historical artifacts and cultural scenes, with a resolution of $1920 \times 1080$, suitable for high-quality 3D reconstruction. The Bootstrap3D synthetic dataset contains millions of multi-view images covering creative objects such as fictional creatures and cultural symbols.

The specific experimental environment and parameter configuration are shown in Table 1. Based on this experimental environment, the study compared the introduction of raw NeRF [16,28], NeRF based on multi-resolution texture pyramid (Mip-based, Mip-NeRF) [2], Instant Neural Graphics Primitives with a multi-resolution hash encoding (Instant-NGP) [17], and INeRF model proposed in this paper.

Firstly, using Peak Signal to Noise Ratio (PSNR) as a comparison metric, tests were conducted on different datasets, and the results are shown in Figure 7, where the PSNR comparison performance of four 3D reconstruction models on two datasets are displayed. In Figure 7a, on the UoMCult3D dataset, NeRF had the weakest performance with a PSNR of 25.82 at the 500th iteration. Mip-NeRF and Instant-NGP reached 28.19 and

Tab. 1. Experimental environment and parameter configuration.

| Type | Name | Version |
|---|---|---|
| Hardware equipment | CPU | Intel Xeon Gold 6248R, 3.0 GHz, 24C |
| | GPU | NVIDIA RTX 3090, 24 GB RAM |
| | RAM | 128 GB DDR4 |
| | Memory device | 2 TB NVMe SSD |
| Software equipment | Operating system | Ubuntu 20.04 LTS |
| | DL framework | PyTorch 1.13 |
| | Graphics rendering | Unity 2022.3 (HDRP line pipe) |
| | AR develop | ARCore 1.35, ARKit 5.0 |
| | VR develop | SteamVR 2.0, OpenXR 1.0 |
| Parameter name | Learning rate | 0.001 |
| | Batch size | 1024 |
| | Render resolution | $800 \times 800$ pixels |
| | Real-time render target frame rate | $\geq 30$ FPS |

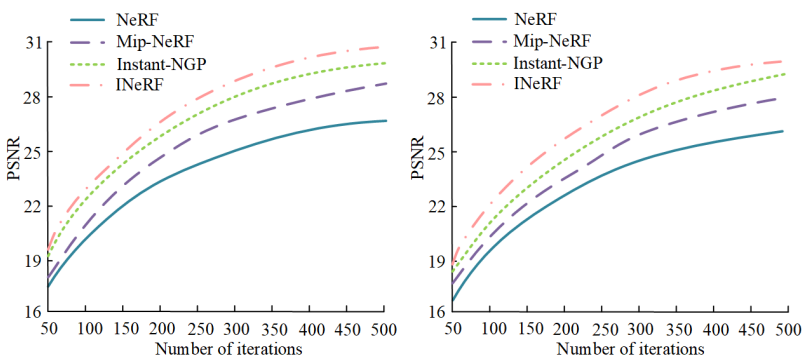


Fig. 7. PSNR comparison of four models with different data sets: (**a**) UoM-Culture3D, (**b**) Bootstrap3D.

30.11, respectively, while INeRF performed the best, stabilizing at 30.63, with an average improvement of 9.24% compared to the other three models. In Figure 7b, INeRF still had a significant advantage on the Bootstrap3D dataset, with a PSNR of 30.15 at the 500th iteration, an average increase of 8.17% compared to other models. This indicated that INeRF had good universality and reconstruction stability in stylized data and cultural images. On this basis, the graphic loading speed and Root Mean Square Error (RMSE) of four models on the AR platform were tested, and the results are shown in Figure 8. According to Figure 8a, as the number of experiments increased, the loading speed of

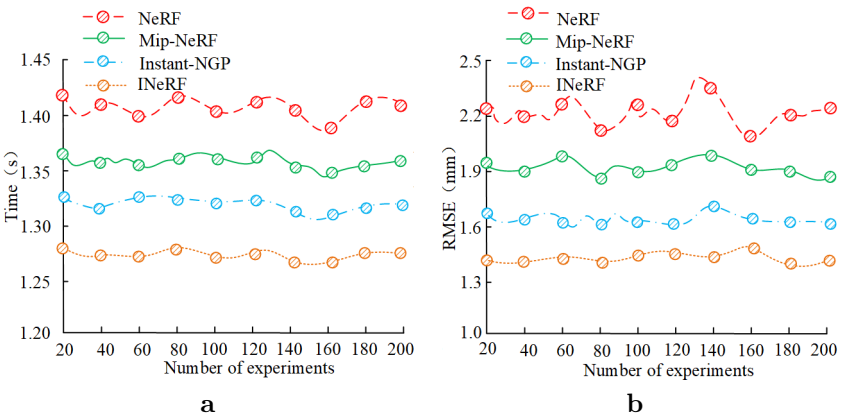


Fig. 8. Comparison of parameters of four models: (**a**) graphic loading time, (**b**) RMSE.

the INeRF model remained at a relatively low level of about 1.26 s, demonstrating high stability and efficiency. In contrast, the NeRF model had the longest loading time, close to 1.41 seconds, and it fluctuated greatly. This might have been due to its reliance on a large number of input images and scene-by-scene optimization training, which resulted in high computational complexity and a slow speed during the loading process. Based on Figure 8b, INeRF had the lowest RMSE value among 200 experiments, stabilizing at around 1.42 mm, with an average reduction of 25.28% compared to other models. Overall, the balance between speed and accuracy of INeRF validated the effectiveness of its improved architecture, providing a reliable technical path for high-fidelity digitization of cultural heritage.

Meanwhile, the Structural Similarity Index Measure (SSIM) of four models on different datasets were compared, and the results are shown in Figure 9. Figure 9a shows the SSIM comparison of four models on the UoM-Culture3D dataset. As the number of iterations increased, the SSIM value of INeRF gradually rose and tended to stabilize. When the number of iterations reached 500, the SSIM value of INeRF remained stable at around 0.88, significantly better than the other three models. Figure 9b presents the SSIM comparison of four models for the Bootstrap3D dataset. NeRF performed better than other models on the Bootstrap3D dataset. When the number of iterations reached 500, the SSIM value of INeRF reached 0.89. This indicates that INeRF can effectively integrate geometric features of different scales, enhancing the model's perception and reconstruction ability of complex image structures.

To directly validate the accuracy of the INeRF algorithm in 3D structure reconstruction, a quantitative evaluation based on point cloud comparison was conducted on the

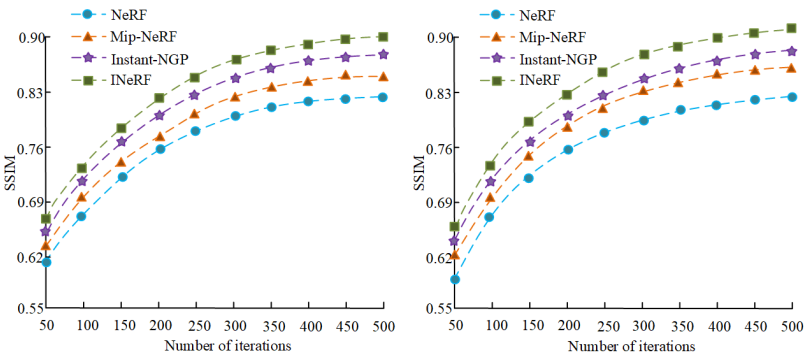


Fig. 9. Comparison of SSIM of four models for different data sets: (**a**) UoM-Culture3D, (**b**) Bootstrap3D.

Tab. 2. Comparison of 3D geometric reconstruction effects of different 3D reconstruction techniques. Asterisks '*' and '**' indicate statistically significant differences compared to INeRF at $p < 0.05$ and $p < 0.01$, respectively.

| Model | NeRF | Mip-NeRF | Instant-NGP | INeRF | 3DGS | DiffRF |
|---|---|---|---|---|---|---|
| Chamfer dist. [mm] | 2.12* | 1.88* | 1.55* | 1.42 | 1.60 | 1.50 |
| Hausdorff dist. [mm] | 6.48** | 5.92** | 5.11* | 4.78 | 5.05 | 4.92 |
| F1-score @0.05 | 0.42** | 0.51** | 0.61* | 0.65 | 0.62 | 0.63 |
| F1-score @0.1 | 0.59** | 0.65** | 0.74* | 0.78 | 0.75 | 0.76 |
| F1-score @0.2 | 0.71** | 0.76** | 0.83* | 0.86 | 0.84 | 0.85 |
| Normal Consistency | 0.78** | 0.81** | 0.85* | 0.88 | 0.86 | 0.87 |
| Training time [h] | 12.42 | 9.71 | 4.15 | 5.31 | 6.27 | 6.82 |
| Peak vRAM [GB] | 18.60 | 16.24 | 9.83 | 11.41 | 12.58 | 13.16 |

UoM-Culture3D dataset. Two emerging 3D reconstruction techniques were also introduced for comparison: the 3D Gaussian Splatting (3DGS) model and the Rendering-Guided 3D Radiance Field Diffusion Model (DiffRF). Marching Cubes algorithm was used to extract meshes from the density fields predicted by each model, and 50 000 vertices were uniformly sampled to generate point clouds for evaluation. The results are shown in Table 2. It can be seen that the NeRF model performs the worst in various indicators, reflecting its insufficient ability to reconstruct complex textures and details in sparse views, as well as high resource requirements. Mip NeRF improved feature expression through multi-resolution texture pyramids, reducing Chamfer Distance to

1.88 mm and Hausdorff Distance to 5.92 mm. However, there were still significant differences ($p < 0.05$) between the improvements and INeRF. Instant NGP further optimized the point cloud distribution under dense feature encoding, with a Chamfer Distance of 1.55 mm and a normal consistency of 0.85. Although the overall accuracy is close, the difference with INeRF is still significant ($p < 0.05$). In contrast, INeRF achieved the best performance on all indicators, with the lowest Chamfer Distance being 1.42 mm, the Hausdorff Distance dropping to 4.78 mm, F1-scores reaching 0.78 and 0.86 at the 0.1 and 0.2 thresholds, respectively, and a normal consistency of 0.88. The INeRF maintains high accuracy while controlling the training time to 5.31 h, with a video memory usage of only 11.41 GB. Although slightly higher than Instant NGP, it still demonstrates good deployability in resource constrained environments, reflecting the balance advantage between accuracy and efficiency. The difference from most methods is significant or highly significant, thanks to the collaborative optimization of multi-level cost volume fusion and Gaussian uniform mixture sampling strategy in details and global structure. For emerging technologies, the 3D Gaussian jet model and rendering guided radiation field diffusion model approach Instant NGP on Chamfer Distance and F1-score, with no significant difference compared to INeRF, but slightly lower in performance, indicating that there are still subtle geometric errors in sparse input and complex texture scenes.

Based on various indicators and statistical analysis, INeRF exhibits excellent performance in point cloud accuracy, surface normal consistency, and F1-score at different scales. It also shows strong advantages in computational resource utilization, verifying its robustness and reliability in high-precision 3D reconstruction. At the same time, it demonstrates strong adaptability to complex textures and geometric structures in the process of cultural heritage digitization.

### 4.2. Visual communication effect verification

To validate the effectiveness of the proposed visual communication strategy integrating AR and VR in actual deployment, the study conducted on-site deployment tests in museum exhibition spaces. The deployment included: AR end: Using ARCore/ARKit devices to scan the exhibition area, accurately anchor the location of exhibits, and overlay virtual information. VR end: Using SteamVR devices to construct virtual exhibitions of historical scenes, allowing users to freely interact in the virtual space. The actual measurement data covers indicators such as spatial perception consistency, interaction fluidity, immersion, and cultural understanding perception (out of 10 points) for 30 test subjects. The study compared the traditional 2D display, single VR, and single AR technologies with the proposed fusion strategy, and tested the spatial perception consistency and average dwell time of the four technologies in four cultural and creative scenes: porcelain, murals, ancient architecture, and bronze ware. The results are shown in Figure 10.

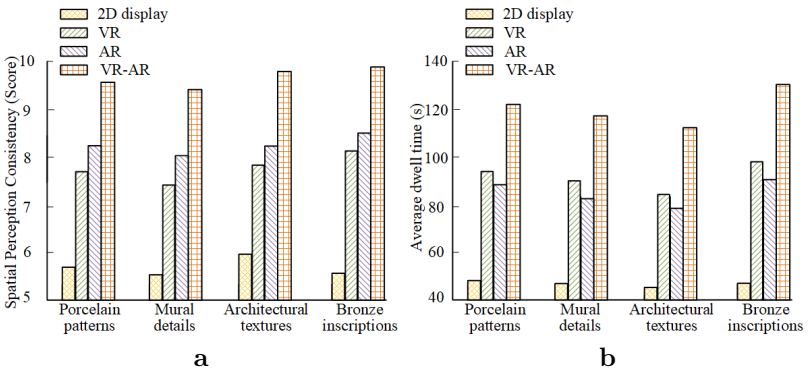


Fig. 10. Comparison of visual communication effects in different cultural and creative scenes: (**a**) spatial consistency ratings; (**b**) average dwell time.

According to Fig. 10a, the scores for integrating VR–AR technology in the four cultural and creative scenes of porcelain, mural, ancient architecture, and bronze ware were 9.50 points, 9.40 points, 9.60 points, and 9.83 points, respectively. When compared with the other three single visual communication technologies, the average scores had increased by 32.72%, 38.35%, 32.27%, and 34.25%, respectively. This indicated that the integration of VR–AR technology achieved better real-world mapping and spatial positioning of three-dimensional structures under the fusion of virtual and real environments. Meanwhile, based on Fig. 10b, the average dwell time of the fusion strategy in the four cultural and creative scenes of porcelain, mural, ancient architecture, and bronze ware was 121.24 s, 118.16 s, 115.67 s, and 130.13 s, respectively. This was an average increase of 59.67%, 58.32%, 57.19%, and 62.25% compared to the other three technologies. By constructing an immersive virtual space and implementing a personalized interactive content push mechanism, users were able to form a deeper sense of participation and cultural context immersion during the experience, which in turn extended their stay time.

Finally, the interactive experience and cultural perception effects of visual communication technology integrating AR and VR were studied, and the results are shown in Table 3. Visual communication technology that integrates VR and AR significantly outperforms single 2D display, VR, or AR solutions in terms of scene detail recognition, interaction fluidity, visual immersion, cultural compatibility, and memory retention. Most of these differences are highly significant ($p < 0.01$), confirming its advantages. Specifically, the scene detail recognition accuracy of the proposed fusion VR and AR visual communication technology reached 92.36%, an average improvement of 23.37% compared to the other three methods, indicating that it had higher accuracy in visual clarity and spatial recognition. In terms of interaction fluency and visual immersion,

Tab. 3. Comparison of interactive experience and cultural perception effect of different visual communication technologies. Asterisks '*' and '**' indicate statistically significant differences compared to INeRF at $p < 0.05$ and $p < 0.01$, respectively.

| Index | Scene detail recognition accuracy [%] | Interaction fluency [points] | Visual immersion [part] | Cultural fit [%] | Memory retention [%] |
|---|---|---|---|---|---|
| 2D display | 64.37** | 5.18** | 4.92** | 61.25** | 58.63** |
| VR | 78.45** | 7.86* | 7.32** | 76.12** | 71.40** |
| AR | 81.78* | 7.12** | 8.47* | 80.56* | 74.93* |
| VR–AR | 92.36 | 9.14 | 9.68 | 90.42 | 86.71 |

the fusion strategy achieved scores of 9.14 and 9.68, respectively, with an average improvement of 36.07% and 39.94% compared to the other three methods. This indicated that it had advantages in operational response and system feedback, while also providing a more immersive cultural experience. In addition, the cultural fit and memory retention of fusion technology were 90.42% and 86.71%, respectively, with an average improvement of 24.47% and 26.92%, indicating that it was more accurate in conveying cultural connotations and symbol fit, and had a stronger effect on retaining cultural information. Overall, the integration of VR and AR technology had significant advantages in enhancing user immersion, improving cultural understanding and memory retention, which validated the scientific and practical nature of the visual communication strategies proposed in the study.

## 5. Discussion

The research is dedicated to addressing the challenge of synergistically optimizing high-precision reconstruction and cross-platform immersive communication in the digitization of cultural heritage. While 3D reconstruction technology has made progress in multiple fields, it still faces limitations in scene adaptability: a new real-time 3D reconstruction framework significantly enhances maritime situational awareness by integrating temporal 2D video data. Its optimized dynamic reconstruction pipeline enables real-time computation on GPU-accelerated embedded devices. However, it lacks the ability to predict the pose of semi-static objects, making it difficult to capture the geometric continuity of cultural relics under micro-movement conditions [20]. Visual tracking technology based on real-time localization and mapping serves as the core support for augmented reality localization. While it can real-time obtain user pose information, it faces inherent limitations in static scenes due to global localization drift and translation dependency, leading to insufficient spatial anchoring stability in cultural heritage sites [1]. In the field of medical imaging, three-dimensional reconstruction methods for brain tumors based

on magnetic resonance imaging demonstrate efficient and precise visualization capabilities. However, when faced with the multi-layered composite texture structure of cultural relics, their topological adaptability remains weak [8]. The aforementioned technologies are either constrained by the integrity of dynamic modeling, limited by the robustness of static localization, or lack the generalization capability for heterogeneous structures, and thus fail to bridge the dual demands of millimeter-level precision reconstruction and multi-modal immersive narrative in cultural heritage digitization.

Therefore, this study aims to establish an integrated system that combines high-precision digital reconstruction with immersive cultural communication, proposing the INeRF algorithm and a multi-dimensional design method that integrates AR and VR technologies. By introducing a multi-level cost-volume fusion module, it achieves collaborative optimization of geometric features across scales, and adopts a Gaussian-uniform hybrid sampling strategy to enhance computational efficiency. Additionally, it combines AR and VR technologies to construct a three-tier communication system encompassing data generation, visual expression, and interactive experience. At the technical implementation level, the system uses multi-sensor fusion to achieve real-time positioning and environmental perception. It also uses dynamic lighting matching, object posture adjustment, and content stream optimization to ensure the accurate presentation of virtual objects on different platforms and in different exhibition environments. Finally, the study validated the feasibility of the integrated AR–VR strategy through field deployment. Field tests demonstrated that the system could achieve stable virtual overlay and multimodal interaction in real exhibition spaces, and user feedback showed significant improvements in cultural information understanding and immersive experiences.

It should be noted that there are still certain limitations in the experimental and validation of the research. Firstly, the test object mainly focuses on the 3D reconstruction of static scenes. However, with the continuous expansion of digital demand for cultural heritage, dynamic cultural heritage such as dance, ceremony, and performance have gradually become research hotspots. For scenes with temporal variability, relying solely on static modeling cannot fully capture their temporal features and dynamic details. Secondly, there are certain limitations to the user research conducted. The current experiment only involves 30 participants, with a relatively limited sample size and a relatively small group composition, which may affect the universality of the research conclusions to some extent and not fully reflect the real experiences of users with different backgrounds.

Future research will further expand the applicability of the INeRF framework in dynamic modeling, such as by introducing temporal consistency constraints and combining optical flow or skeleton driven motion modeling methods to achieve high fidelity reconstruction and presentation of dynamic cultural heritage. At the same time, it is necessary to expand the sample size in user research, increase the dual participation of experts in

cultural heritage protection and ordinary visitors, in order to obtain a more comprehensive evaluation. With further validation of the system in multi-user collaboration and dynamic exhibition scenarios, its universality and sustainability in digital protection and cross platform dissemination of cultural heritage are expected to be greatly improved.

## 6. Conclusion

Compared with existing methods, the INeRF based method improves reconstruction accuracy by 9%, reduces RMSE to 1.42 mm, and enhances visual immersion by nearly 40%. AR–VR integration significantly enhances cultural detail recognition and user engagement. Although research still has limitations in terms of static scene adaptability and small user sample size, future work will explore lightweight network architectures and broader user testing to achieve more universal applications and higher dynamic scene adaptability.

## Funding

This paper was supported by the 2024 Guangxi University Research Foundation Capacity Enhancement Project for Middle-aged and Young Teachers: *Research on the Collection, Organization and Digital Inheritance of Endangered Materials on the Feitao Art of Maonan Ethnic Group in Guangxi* (No. 2024KY0765).

## Conflicts of Interest

The authors declare no conflict of interest.

## Data Availability

The data supporting the findings of this study are referenced in the literature.

## References

[1] L. Baker, J. Ventura, T. Langlotz, S. Gul, S. Mills, et al. Localization and tracking of stationary users for augmented reality. *The Visual Computer* 40(1):227–244, 2024. doi:10.1007/s00371-023-02777-2.

[2] J. T. Barron, B. Mildenhall, M. Tancik, P. Hedman, R. Martin-Brualla, et al. Mip-NeRF: A multiscale representation for anti-aliasing neural radiance fields. In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5835–5844, 2021. doi:10.1109/ICCV48922.2021.00580.

[3] J. Bast. Managing the image. The visual communication strategy of European right-wing populist politicians on Instagram. *Journal of Political Marketing* 23(1):1–25, 2024. doi:10.1080/15377857.2021.1892901.

[4] J.-J. Cao, S.-M. Fang, and H. Contreras. Multimodal fusion visual communication method based on genetic algorithm. *Journal of Network Intelligence* 10(2):1071–1083, 2025. https://bit.kuas.edu.tw/~jni/2025/vol10/s2/34.JNI-S-2024-05-019.pdf.

[5] J. Fang and X. Gong. Application of visual communication in digital animation advertising design using convolutional neural networks and big data. *Peerj Computer Science* 9:e1383, 2023. doi:10.7717/peerj-cs.1383.

[6] FREEPIK. Find icons that go together. Fast. https://www.freepik.com/icons.

[7] Y. Ge, B. Guo, P. Zha, S. Jiang, Z. Jiang, et al. 3D reconstruction of ancient buildings using UAV images and neural radiation field with depth supervision. *Remote Sensing* 16(3):473, 2024. doi:10.3390/rs16030473.

[8] M. A. Guerroudji, K. Amara, M. Lichouri, N. Zenati, and M. Masmoudi. A 3D visualization-based augmented reality application for brain tumor segmentation. *Computer Animation and Virtual Worlds* 35(1):e2223, JAN 2024. doi:10.1002/cav.2223.

[9] A. Houdard, A. Leclaire, N. Papadakis, and J. Rabin. A generative model for texture synthesis based on optimal transport between feature distributions. *Journal of Mathematical Imaging and Vision* 65(1):4–28, 2023. doi:10.1007/s10851-022-01108-9.

[10] Z. Jia, B. Wang, and C. Chen. Drone-nerf: Efficient nerf based 3D scene reconstruction for large-scale drone survey. *Image and Vision Computing* 143:104920, 2024. doi:10.1016/j.imavis.2024.104920.

[11] X. Liao, X. Wei, M. Zhou, and S. Kwong. Full-reference image quality assessment: Addressing content misalignment issue by comparing order statistics of deep features. *IEEE Transactions on Broadcasting* 70(1):305–315, 2023. doi:10.1109/TBC.2023.3294835.

[12] J. Lin, G. Sharma, and T. N. Pappas. Toward universal texture synthesis by combining texton broadcasting with noise injection in StyleGAN-2. *e-Prime – Advances in Electrical Engineering, Electronics and Energy* 3:100092, 2023. doi:10.1016/j.prime.2022.100092.

[13] F. Liu, B. Lin, and K. Meng. Design and realization of rural environment art construction of cultural image and visual communication. *International Journal of Environmental Research and Public Health* 20(5):4001, 2023. doi:10.3390/ijerph20054001.

[14] W. Liu, Y. Zang, Z. Xiong, X. Bian, C. Wen, et al. 3D building model generation from MLS point cloud and 3D mesh using multi-source data fusion. *International Journal of Applied Earth Observation and Geoinformation* 116:103171, 2023. doi:10.1016/j.jag.2022.103171.

[15] G. Mazzacca, A. Karami, S. Rigon, E. Farella, P. Trybala, et al. Nerf for heritage 3D reconstruction. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 48(M-2-2023):1051–1058, 2023. doi:10.5194/isprs-archives-XLVIII-M-2-2023-1051-2023.

[16] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, et al. NeRF: representing scenes as neural radiance fields for view synthesis. *Communications of the ACM* 65(1):99–106, 2021. doi:10.1145/3503250.

[17] T. Müller, A. Evans, C. Schied, and A. Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics* 41(4):102, 2022. doi:10.1145/3528223.3530127.

[18] M. Pepe, V. S. Alfio, and D. Costantino. Assessment of 3D model for photogrammetric purposes using AI tools based on NeRF algorithm. *Heritage* 6(8):5719–5731, 2023. doi:10.3390/heritage6080301.

[19] S. Qiu, S. Wang, X. Chen, F. Qian, and Y. Xiao. Ship shape reconstruction for three-dimensional situational awareness of smart ships based on neural radiation field. *Engineering Applications of Artificial Intelligence* 136:108858, 2024. doi:10.1016/j.engappai.2024.108858.

[20] F. Sattler, B. Carrillo-Perez, S. Barnes, K. Stebner, M. Stephan, et al. Embedded 3D reconstruction of dynamic objects in real time for maritime situational awareness pictures. *The Visual Computer* 40(2):571–584, 2024. doi:10.1007/s00371-023-02802-4.

[21] S. Shen, S. Xing, X. Sang, B. Yan, and Y. Chen. Virtual stereo content rendering technology review for light-field display. *Displays* 76:102320, 2023. doi:10.1016/j.displa.2022.102320.

[22] X. Shi and R. Villegas. AI technology in the virtual reality environment of graphic design of dynamic art visual communication frame. *Journal of Computational Methods in Sciences and Engineering* 25(3):2603–2616, 2025. doi:10.1177/14727978251321333.

[23] Z. Sun. BS-Objaverse. Hugging Face. `https://huggingface.co/datasets/Zery/BS-Objaverse/`.

[24] Z. Sun, T. Wu, P. Zhang, Y. Zang, X. Dong, et al. Bootstrap3D: Improving multi-view diffusion model with synthetic data. arXiv, arXiv:2406.00093v2, 2024. doi:10.48550/arXiv.2406.00093.

[25] Xinyi_Zheng. CULTURE3D: Cultural Landmarks and Terrain Dataset for 3D Applications. GitHub. `https://github.com/X-Intelligence-Labs/CULTURE3D`.

[26] V. O. Yachnaya, V. R. Lutsiv, and R. O. Malashin. Modern automatic recognition technologies for visual communication tools. *Computer Optics* 47(2):287–305, 2023. doi:10.18287/2412-6179-CO-1154.

[27] 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.

[28] J.-W. Yang, J.-M. Sun, Y.-L. Yang, J. Yang, Y. Shan, et al. DMiT: Deformable Mipmapped Triplane representation for dynamic scenes. In: *Computer Vision – ECCV 2024*, pp. 436–453. Springer Nature Switzerland, Cham, 2025. doi:10.1007/978-3-031-73001-6_25.

[29] J. You and X. Lu. Visual communication design based on machine vision and digital media communication technology. *KSII Transactions on Internet & Information Systems* 19(6):1888–1907, 2025. doi:10.3837/tiis.2025.06.007.

[30] S. H. Yudhanto, F. Risdianto, and A. T. Artanto. Cultural and communication approaches in the design of visual communication design works. *Journal of Linguistics, Culture and Communication* 1(1):79–90, 2023. doi:10.61320/jolcc.v1i1.79-90.

[31] 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*, pp. 7523–7532, 2024. doi:10.1145/3664647.3680777.

[32] M. Zhao. Application of image reconstruction algorithm combining FCN and Pix2Pix in visual communication design. *Journal of Computational Methods in Sciences and Engineering* 25(4):3137–3151, 2025. doi:10.1177/14727978251319398.

# A. Appendix

**Pseudocode of INeRF algorithm**

INeRF algorithm

Input:
$I_{\text{src}}$
$K$: Camera intrinsic matrix
Output:
$M$: High-precision 3D mesh model (geometry + texture)
// Step 1: Multi-view generation (replaces multi-image input)
$I_{\text{views}} \leftarrow \text{MultiViewGenerator}(I_{\text{src}})$ //Generate $N$ virtual views $\{I_1, I_2, ..., I_N\}$
$\Theta_{\text{cam}} \leftarrow \text{EstimateCameraPoses}(I_{\text{views}}, K)$ //Estimate virtual view poses
//Step 2: Geometric reasoning (multi-level cost volume fusion)
$F_{\text{multi}} = [\,]$ //Initialize multi-scale feature list
for each $I_i$ in $I_{\text{views}}$
  for each scale $s$ in $[1, 2, 4]$//Multi-resolution feature extraction
    $F_s \leftarrow \text{CNN\_Encoder}(I_i, \text{scale} = s)$ //Extract features at scale $s$
    $F_{\text{multi}}[s] \leftarrow F_s$
  end for
  $C_i \leftarrow \text{BuildCostVolume}(F_{\text{multi}}, \Theta_{\text{cam}}[i])$ //Construct cost volume for view $i$
end for
$F_{\text{fused}} \leftarrow \text{MultiLevelFusion}(C_{\text{all}})$ //Fuse cost volumes (Eq. (4), Fig. 3)
//Step 3: Neural radiance field modeling
for each pixel $p$ in target view:
  ray $r \leftarrow \text{GenerateRay}(p, \Theta_{\text{cam\_target}})$
  //Gaussian-uniform hybrid sampling (Eq. (5))
  samples $\leftarrow \text{GaussUniformHybridSampling}(r, \text{depth\_prior=DepthMap}(F_{\text{fused}}),$
    $\mu = \text{depth\_mean}, \sigma = 0.2, \alpha = 0.7)$ //$\alpha$: Gaussian sampling weight
  $\sigma, c \leftarrow [\,]$ //Store density and color
  for each sample point $x$ in samples
    $\text{feat}_{3d} \leftarrow \text{Query3DFeature}(x, F_{\text{fused}})$ //Query 3D local feature
    $\text{feat}_{\text{dir}} \leftarrow \text{Encode}(\text{view\_dir})$ //View direction encoding
    $(\sigma_x, c_x) \leftarrow \text{MLP}_{\sigma c}(\text{feat}_{3d}, \text{feat}_{\text{dir}})$ //Predict density and color
    $\sigma.\text{append}(\sigma_x)$; $c.\text{append}(c_x)$
  end for
  //Volume rendering (Eq. (2))
  $\hat{C}_p \leftarrow \text{VolumeRendering}(\sigma, c, \text{samples})$
  $\hat{D}_p \leftarrow \text{DepthMapRendering}(\sigma, \text{samples})$ //Predict depth map
end for
//Step 4: Self-supervised optimization
$L_{\text{rgb}} \leftarrow \text{MSE}(\hat{C}, I_{\text{gt}})$ //RGB rendering loss (Eq. (3))
$L_{\text{depth}} \leftarrow \text{DepthConsistencyLoss}(\hat{D}, \text{FusedDepth})$ //Depth self-supervised loss
$L_{\text{total}} \leftarrow \lambda_1 L_{\text{rgb}} + \lambda_2 L_{\text{depth}}$ //$\lambda_1 = 1.0$, $\lambda_2 = 0.5$ (tunable)
Update $\text{MLP}_{\sigma c}$ via $\nabla L_{\text{total}}$ //Backpropagation update
//Step 5: High-res texture generation & model simplification
$M_{\text{highres}} \leftarrow \text{TextureSynthesis}(F_{\text{fused}}, \text{MLP}_{\sigma c})$ //Generate textured dense mesh
$M \leftarrow \text{MeshSimplification}(M_{\text{highres}}, \text{target\_faces=50k})$ //Simplify model
return $M$

# Intelligent Extraction and Layout Optimization of Digital Media Visual Elements Based on Computer Vision

Hebin Wu*
*Department of Computer Engineering, Shanxi Engineering Vocational College, Taiyuan, China*
**Corresponding author: Hebin Wu (WuHebin1989@163.com)*

**Submitted:** 04 Jun 2025 **Accepted:** 09 Dec, 2025 **Published:** 21 Feb, 2026 

**Abstract** In the field of digital media, intelligent extraction and layout optimization of visual elements face challenges such as inaccurate semantic understanding of elements and low efficiency in generating layout strategies. This study proposes an extraction and layout optimization model that integrates visual semantic understanding with intelligent optimization strategies, based on a segmentation Vision Transformer and Multi-Objective Firefly Algorithm. The model also utilizes the improved optical flow methods to efficiently capture dynamic information during the design process. Experimental results show that the segmentation Vision Transformer algorithm achieves an extraction accuracy of $98.8 \pm 0.2\%$ for different categories of visual elements. As the training progresses to 50 iterations, the average Intersection-Over-Union stabilizes at 0.95, and the harmonic mean of recall reaches $98.17 \pm 0.38\%$. The evaluation of the integrated model shows that it achieves 99% accuracy in extracting visually similar elements. After layout optimization using the model, the aesthetic score increases to 95.6, and the spatial occupancy rate improves to 97.2%. The above results indicate that the model proposed by the research institute can effectively enhance the accuracy of visual element extraction and the quality of layout optimization, significantly reducing the reliance of traditional methods on manual rules, and providing an efficient and adaptive solution for the automated design of digital media.

**Keywords:** digital media, layout optimization, SAM, ViT, PWCNet, MOFA.

## 1. Introduction

In recent years, with the rapid development of the digital media industry, users have raised higher demands for the accuracy and efficiency of visual element processing. The application of computer vision algorithms in the field of digital media not only enables precise extraction of visual elements but also enhances the visual appeal of content through layout optimization [16]. Therefore, research on intelligent extraction and layout optimization algorithms for visual elements is of great significance for technological innovation in the digital media industry. Currently, mainstream visual processing methods have limitations, including poor adaptability to complex scenes, weak dynamic element processing capabilities, and a lack of multi-objective coordination in layout optimization [28]. Traditional methods are prone to incomplete extraction when dealing with dynamic elements in videos, and the layout is also difficult to balance aesthetics and functionality. Specifically, the core research issues that urgently need to be addressed in the current field can be summarized into three points. The first is the disconnection between *segmentation and semantics* in the extraction of static visual elements. Although

existing segmentation algorithms can accurately locate the boundaries of elements, they are difficult to capture the semantic associations between elements and cannot directly support layout optimization. Second, the temporal correlation modeling of dynamic visual elements is insufficient. Traditional methods are difficult to accurately calculate the inter-frame trajectories of dynamic elements in fast-moving or occluded scenes, and cannot provide spatio-temporal consistency features, which easily leads to chaotic dynamic layout. Thirdly, there is a lack of layout optimization and dynamic adaptation of element features. Most existing layout algorithms are based on fixed rules for optimization and do not take dynamic and semantic features of elements as constraints, resulting in poor adaptability to multiple scenarios and difficulty in balancing aesthetics and functionality. Compared with traditional algorithms, the Segment Anything Model (SAM) has image segmentation and zero-shot generalization capabilities, and can efficiently extract static visual elements [27]. The Vision Transformer (ViT), based on the Transformer architecture, can effectively capture semantic relationships between elements [22]. When combined with the improved optical flow algorithm PWCNet [23], it can accurately calculate the motion trajectories of dynamic elements, improving extraction accuracy in dynamic scenes. In terms of layout optimization, the improved Multi-Objective Firefly Algorithm (MOFA) simulates collective search behavior to simultaneously optimize multiple objectives, such as element position, proportion, and color, balancing aesthetics and information delivery efficiency [19]. As a result, this study proposes a digital media visual element extraction and layout model that integrates SAM, ViT, PWCNet, and MOFA. The first three algorithms enable accurate element extraction, while MOFA performs intelligent layout optimization. Finally, the extraction and layout modules are deeply integrated. Specifically, the innovation points of the research are reflected in three aspects. First, a *semantically – dynamic* dual-driven extraction mechanism is constructed. Through the cross-layer feature fusion of SAM and ViT, the pixel-level segmentation results are deeply bound with global semantic associations, solving the problem of the disconnection between element extraction and semantic understanding in traditional methods. Second, a layout optimization framework under dynamic constraints was designed. For the first time, the optical flow field output by PWCNet was used as a hard constraint condition for MOFA, enabling the layout optimization process to respond in real time to the movement trajectories of dynamic elements and breaking through the adaptation limitations of static layout algorithms to dynamic scenes. Thirdly, a modular collaborative learning strategy is proposed. Through the parameter mutual transmission mechanism between the extraction module and the layout module, an end-to-end optimization of *extraction accuracy – layout quality* is achieved, avoiding the problem of error accumulation in the traditional series model. These innovative designs enable the model to outperform existing single methods or simple combination schemes in terms of complex scene adaptability, dynamic element processing capabilities, and multi-objective coordination and optimization. They provide a more efficient

systematic solution for the digital media scenarios covered by the test, and there is also great potential for its application in a wider range of fields. Subsequently, further testing and optimization will be carried out in combination with technical constraints from more fields.

## 2. Related works

Computer vision, as a technology for machine understanding and interpreting visual information, can extract features, analyze, and recognize image or video data. This mechanism, which simulates human visual perception through algorithms, plays a key role in tasks such as image classification, object detection, and semantic segmentation, and has inspired widespread exploration and in-depth research by scholars worldwide. For example, in the field of obstacle detection, avoidance, and traffic signal and sign recognition, Tan et al. [24] proposed a combination of computer vision and artificial intelligence. Experimental results indicated that computer vision, as a direct entry point for data processing, brought revolutionary changes to future traffic systems and became an indispensable part of autonomous driving. Hassan et al. [6], in response to the optimization and improvement of computer vision task models, proposed a stochastic gradient descent machine learning optimization algorithm. Testing on the ISIC standard dataset showed that the optimizer significantly improved the model's performance, with an accuracy of 97.30%. Li et al. [14], addressing the issue of missing 3D models for large numbers of anatomical images and surgical instruments in medical imaging, proposed using MedShapeNet to transform data-driven vision algorithms into medical applications. The results showed that this method helped the medical industry successfully pair over 100 000 medical images with annotations. Blair et al. [1], tackling the issues of low efficiency and high cost in manual specimen classification in biodiversity monitoring, proposed a method that uses computer vision to quickly, automatically, and accurately classify specimen images. Experimental results showed that this method helped ecologists adjust their workflows to achieve research goals. Mahajan et al. [15], aiming to minimize barriers in real-time IoT-enabled robotics applications, proposed a revolutionary framework built with computer vision and deep learning. Compared with state-of-the-art methods, their model improved overall accuracy by about 5%, while reducing computational complexity by 84%.

With the rapid development of digital media technology, accurate element extraction and optimal layout technologies have gradually become core components of the field. Scholars from many countries have conducted in-depth research on these core technologies. Landolsi et al. [12], addressing the cumbersome task of doctors reading information about drugs, diseases, and patients in the medical field, proposed a natural language processing technique to extract useful information and features, focusing on named entity recognition and relationship extraction. The experiments demonstrated

that this technology could effectively assist doctors in extracting information. Zhang et al. [29], aiming to improve information acquisition and extraction efficiency in current intelligent transportation systems, proposed a model that combines artificial intelligence and deep learning to extract real-time traffic information. Experimental results showed that the model had good fitting performance, with an average accuracy above 0.8. Prastyaningtyas et al. [18], in researching the role of information technology in human resource career development, proposed the use of data reduction, visualization, and inference analysis techniques to extract important findings. The study concluded that information technology plays a crucial role in promoting professional growth in human resources. Shen et al. [21], in order to achieve frequent adjustments in the dynamic layout of homepage news content in real-time environments and increase its appeal to readers, proposed a model that combines a hybrid genetic algorithm and local search heuristics. Experiments showed that the model was highly effective in modeling the changing layouts of digital news websites.

In summary, existing research has made certain progress in intelligent extraction and layout optimization. However, these two technologies have not been deeply integrated. The SAM algorithm, which can be combined with ViT, FA algorithms, and optical flow techniques to address the challenges mentioned above, offers a potential solution. Therefore, this study proposes a novel intelligent extraction and layout model that integrates SAM-ViT and MOFA, aiming to improve the efficiency and quality of visual element processing in digital media.

## 3. Intelligent Extraction and Layout Optimization Based on SAM-ViT and Improved FA

### 3.1. Optimization of Vision Transformer Algorithm with SAM

With the rapid development of artificial intelligence and the widespread application of digital media technology, the volume of visual element data has exploded, leading to issues such as low data processing efficiency. Currently, visual element data is scattered across different platforms, constrained by copyright regulations and platform barriers, making data integration difficult and forming data silos [17]. Therefore, this study proposes utilizing the image segmentation and zero-shot generalization capabilities of the SAM algorithm to achieve intelligent extraction of visual elements in digital media. This algorithm can efficiently handle diverse visual data, retaining the value of visual elements while reducing direct dependence on raw data, effectively addressing the problem of data silos. The structure of SAM is shown in Figure 1.

Equation (1) allocates the attention weights among features through the softmax function, enabling the encoder to prioritize focusing on key visual information and enhancing the segmentation accuracy. SAM first inputs the target image. The image is

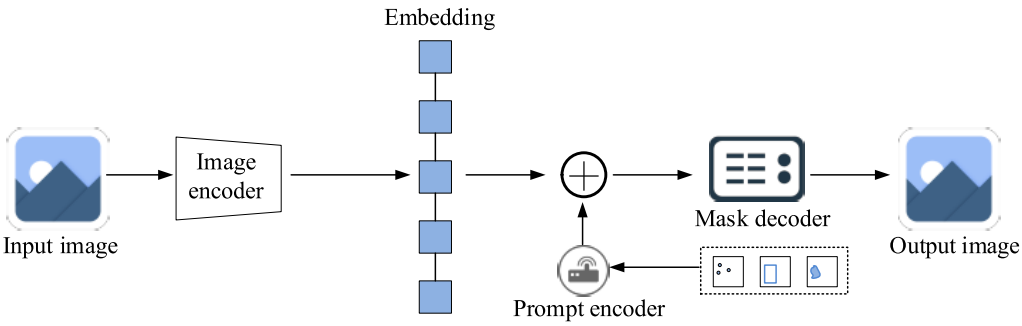


Fig. 1. SAM structure diagram.

processed by an image encoder to generate embedded features, while the prompt encoder processes inputs such as points, boxes, and masks. The outputs of both are summed and fed into the mask decoder, which finally outputs the segmentation mask of the image, realizing the image segmentation function. The image encoder transforms the input image into embedded features, and the attention score calculation of the input features is

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^T}{\sqrt{D_k}}\right)V\,, \tag{1}$$

where $Q$ represents the query of the input features, $K$ represents the key of the input features, $V$ represents the value of the input features, and $D_k$ represents the dimension of the query of the input features. The prompt encoder encodes various prompts, and the encoding operation is

$$\begin{cases} P_{\text{emb}}^{\text{sparse}} = \text{SparseEncoder}(p,b)\,, \\ P_{\text{emb}} = \text{Concat}(P_{\text{emb}}^{\text{sparse}}, P_{\text{emb}}^{\text{dense}})\,, \end{cases} \tag{2}$$

where $(p,b)$ represents the coordinates of the hypothetical point prompt, and $P_{\text{emb}}^{\text{dense}}$ and $P_{\text{emb}}^{\text{sparse}}$ represent the dense and sparse prompt embeddings, respectively. The mask decoder combines the image embedding and prompt embedding to predict the segmentation mask. The decoder also uses the Transformer architecture. The input and output operations of the decoder are

$$\begin{cases} X_{\text{decoder}} = \text{Concat}(E, P_{\text{emb}})\,, \\ \hat{M} = \text{Linear}(F_{\text{mask}})\,, \end{cases} \tag{3}$$

where $E$ represents the image embedding, $F_{\text{mask}}$ represents the output mask features from the decoder, which are then processed by a linear layer to obtain the predicted

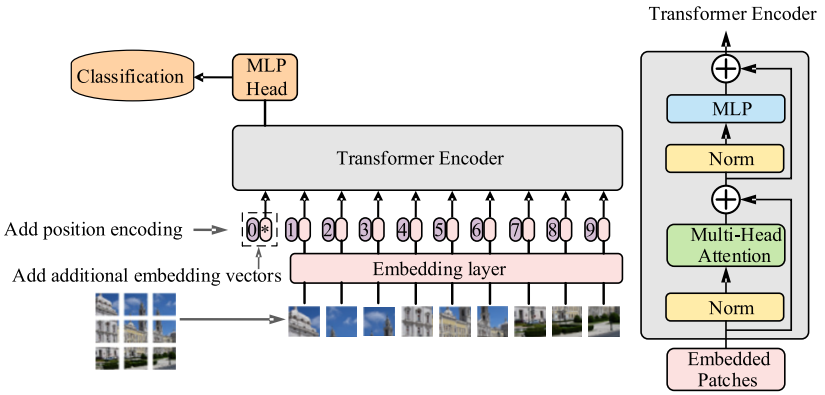


Fig. 2. Structure diagram of the ViT algorithm and its Transformer encoder.

segmentation mask $\hat{M}$. The core advantage of SAM lies in its pixel-level segmentation accuracy. However, its mask decoder can only output the boundary information of elements and lacks the ability to model the semantic associations between elements. In complex scenarios where multiple elements are densely arranged, the problem of *accurate segmentation but semantic fragmentation* is prone to occur. The self-attention mechanism of ViT based on Transformer can capture the long-distance dependencies between elements through global feature interaction, which precisely makes up for the shortcoming of SAM in semantic association modeling [2, 13]. Therefore, the study further introduces the ViT algorithm, leveraging its self-attention mechanism based on the Transformer architecture to effectively capture long-distance dependencies, efficiently model global visual features, and improve the model's representation accuracy in complex scenes to address these limitations. The structure of the ViT algorithm and its Transformer encoder is shown in Figure 2, which shows the ViT algorithm and its Transformer encoder structure. The left side shows the overall flow of ViT. First, the input image is divided into multiple image patches. After linear projection, they are combined with positional embeddings and optional class embeddings. The combined input is then processed by the Transformer encoder, and the classification result is output through the multi-layer perceptron classification head. The right side shows the internal structure of the Transformer encoder. Each layer includes normalization, multi-head attention, residual connections, and a multi-layer perceptron. These components are stacked to encode features. The image is divided into multiple patches, flattened, and projected linearly to obtain embedding vectors. Adding class embeddings and positional encoding, the operation of forming the initial input is given by the equation which solves the problem of no spatial perception in the Transformer:

$$z_0 = (x_{\text{class}}; x_p^1 E; x_p^2 E; \cdots; x_p^N E) + E_{\text{pos}}\,, \tag{4}$$

where $x_{\text{class}}$ represents the class embedding, $x_p^i$ represents the flattened vector of the $i$-th patch, $E$ is the projection matrix, $E_{\text{pos}}$ is the positional encoding vector, and $N$ is the number of patches. The feedforward network performs a nonlinear transformation on the input. The specific operation is

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2\,, \tag{5}$$

where $W_1$ and $W_2$ represent the weight matrices of the two fully connected layers, and $b_1$ and $b_2$ represent the bias vectors of the two fully connected layers. The residual connections after the multi-head self-attention and the residual connections after the feedforward network in the Transformer layer are calculated as

$$z'_\zeta = \text{MSA}(\text{LN}(z_{\zeta-1})) + z_{\zeta-1}\,, z_\zeta = \text{MLP}(\text{LN}(z'_\zeta)) + z'_\zeta\,, \tag{6}$$

where $\zeta$ represents the Transformer layer number, $z_{\zeta-1}$ represents the input vector of the $\zeta-1$-th layer, $\text{LN}(\cdot)$ represents the layer normalization to stabilize the training, $\text{MSA}(\cdot)$ represents the multi-head self-attention, and $\text{MLP}(\cdot)$ represents the multi-layer perceptron. The vector corresponding to the class embedding is extracted. After layer normalization and linear transformation, it is passed through softmax for classification. The operation is as follows

$$y = LN(z_L^0), \quad \text{output} = \text{softmax}(z_L^0 W_{\text{class}})\,, \tag{7}$$

where $z_L^0$ represents the output vector corresponding to the class embedding in the last layer, $W_{\text{class}}$ represents the classification weight matrix, which maps the embedding vector to class probabilities, and softmax represents the activation function, which converts the output into a class probability distribution. The study combines the ViT algorithm with the SAM segmentation algorithm, named SAM-ViT. This combined algorithm enables end-to-end processing from pixel-level segmentation to semantic-level classification, providing high-quality elemental data for subsequent layout optimization. The framework structure of the SAM-ViT algorithm is shown in Figure 3, where it can be seen that the SAM-ViT algorithm first inputs the image for preprocessing, then the SAM segmentation module generates masks. After filtering out noisy masks, region extraction is performed. The extracted regions are input into the ViT module, where feature extraction, encoding, and Transformer processing are done, followed by class prediction through the classification head. For text elements, OCR technology is integrated to optimize the extraction results. Finally, coordinate mapping restores the original image coordinate system, yielding the final output, thus realizing the intelligent extraction of digital media visual elements. Multi-head attention concatenates multiple independent attention outputs and projects them. The operation is

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \cdots, \text{head}_h)W_O\,, \tag{8}$$

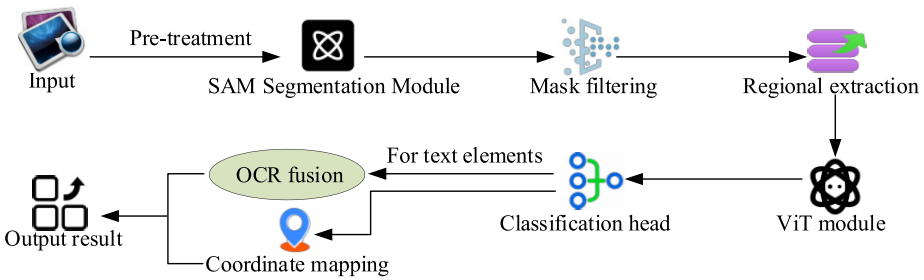


Fig. 3. Framework diagram of the SAM-ViT algorithm.

where $\text{head}_i$ represents the independent attention results, $W_O$ is the projection matrix after concatenation, and $h$ represents the number of heads. The region features after SAM segmentation and the features extracted by ViT are weighted and fused. The operation is as follows:

$$F_{\text{fusion}} = \alpha F_{\text{sam}} + (1-\alpha) F_{\text{vit}} \,. \tag{9}$$

This equation integrates the features of SAM and ViT. The vector $\alpha$ represents the learnable weights, and $F_{\text{sam}}$ and $F_{\text{vit}}$ represent the region features after SAM segmentation and the features extracted by ViT, respectively. The limitations of a single algorithm are addressed by weighting and balancing *pixel-level segmentation accuracy* with *global semantic association*.

### 3.2. Design of Intelligent Extraction and Layout Model Integrating SAM-ViT and MOFA

Although SAM-ViT has solved the problem of *precise segmentation + semantic understanding* of static visual elements, there are a large number of dynamic visual elements in digital media scenarios. The extraction of such elements not only requires spatial features but also temporal motion information. However, SAM-ViT is only for single-frame image processing and lacks the ability to model the temporal correlation of dynamic elements, thus failing to meet the layout optimization requirements of video media or dynamic interactive scenarios. Therefore, the research needs to further introduce dynamic feature capture technology to provide more comprehensive element feature input for the layout model.Therefore, this study proposes optimizing the technology using PWC-Net, which is based on traditional optical flow networks. PWCNet efficiently captures multi-scale optical flow information through its hierarchical feature pyramid structure, and combines a dynamic weight distribution mechanism to enhance tracking capabilities for fast-moving visual elements [5, 8]. It not only provides accurate inter-frame motion feature compensation, improving the spatiotemporal consistency of visual element extraction in dynamic scenes, but also significantly optimizes computational efficiency on

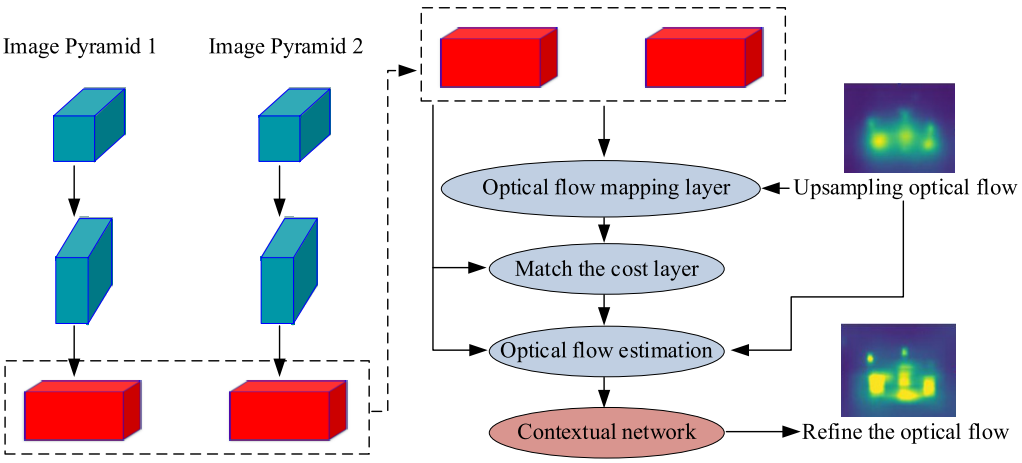


Fig. 4. PWCNet structure diagram.

edge devices. The structure of PWCNet is shown in Figure 4. In this figure it can be seen that PWCNet first constructs two sets of image pyramids to process multi-scale visual features. Optical flow calculation is performed through the optical flow mapping layer, matching cost layer, and optical flow estimation module, followed by upsampling of the optical flow to enhance resolution. Context networks are used to further refine the optical flow results, ensuring that optical flow is efficiently and accurately estimated at different scales, capturing motion information between image sequences. The core computation of PWCNet is

$$f_{\text{flow}} = \text{Decoder}(\text{CostVolume}(F_1, F_2)), \tag{10}$$

where $F_1$ and $F_2$ represent the feature maps of the first and second frame images after deformation by the optical flow field, respectively, and CostVolume$(\cdot)$ represents the similarity cost volume calculation between the two frame feature maps. After intelligent extraction, visual elements may suffer from layout disorder, scattered visual focus, and poor adaptability to multiple scenes. Therefore, the study proposes using MOFA, which effectively coordinates multiple factors of visual elements, to optimize the layout of digital media visual elements.

The structure of MOFA is shown in Figure 5. It first performs population initialization, then calculates the center particles of each subclass. The individual fitness values are updated, followed by the calculation of individual brightness values. After comparing the advantages and disadvantages of individuals, the position of the optimal brightness individual is selected as the updated position. Finally, the algorithm checks whether the target iteration number or precision has been reached. If not, the individual fitness

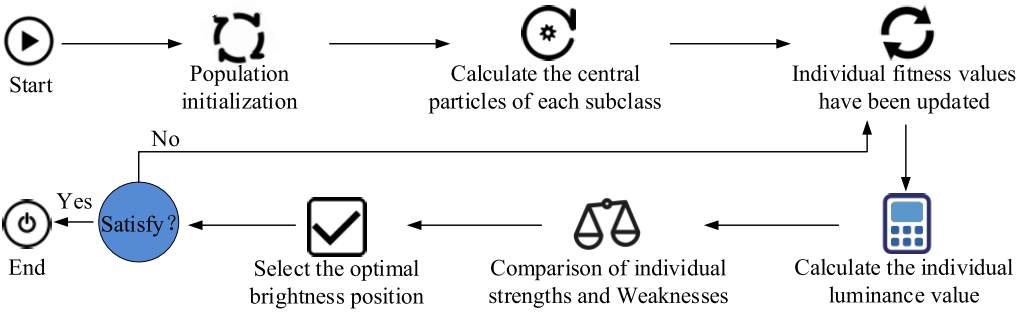


Fig. 5. MOFA structure diagram.

values are updated again. Otherwise, the loop stops, and the result is output. The calculation of individual brightness is

$$I_{ij} = \frac{1}{1 + f_j(x_i)}\,, \tag{11}$$

where $x_i$ represents the position vector of the $i$-th firefly, corresponding to a layout scheme, and $f_j(x)$ represents the $j$-th objective function. The total brightness is a weighted combination of the brightness of each objective, and the specific calculation process is as follows:

$$I_i = \sum_{j=1}^{m} \varpi_j \cdot I_{ij}\,, \tag{12}$$

where $\varpi_j$ represents the weight ratio of each objective. When calculating the brightness value of the firefly, the effect of light intensity attenuation must also be considered:

$$\beta_{ij} = \beta_0 \cdot e^{-\gamma r_{ij}^2}\,, \tag{13}$$

where $\beta_0$ represents the initial attraction, $\gamma$ is the light intensity attenuation coefficient, and $r_{ij}$ represents the Euclidean distance between fireflies $i$ and $j$, the farther the distance, the weaker the attraction. Avoid the algorithm falling into local optimum and ensure the global optimization ability. The firefly movement rule in MOFA and the position update step in which firefly $i$ moves toward the higher brightness $j$ are

$$x_i^{t+1} = x_i^t + \beta_{ij} \cdot (x_j^t - x_i^t) + \alpha \cdot \varepsilon \cdot (u - 1)\,. \tag{14}$$

This equation balances *optimal search* and *random exploration* to enhance the diversity of layout schemes. The variable $x_i^t$ represents the position vector of firefly $i$ in the $t$-th generation, $\alpha$ is the random step length factor, $\varepsilon$ is the random vector, elements follow the $[0, 1]$ uniform distribution, and $u$ is the upper bound of the decision variables.

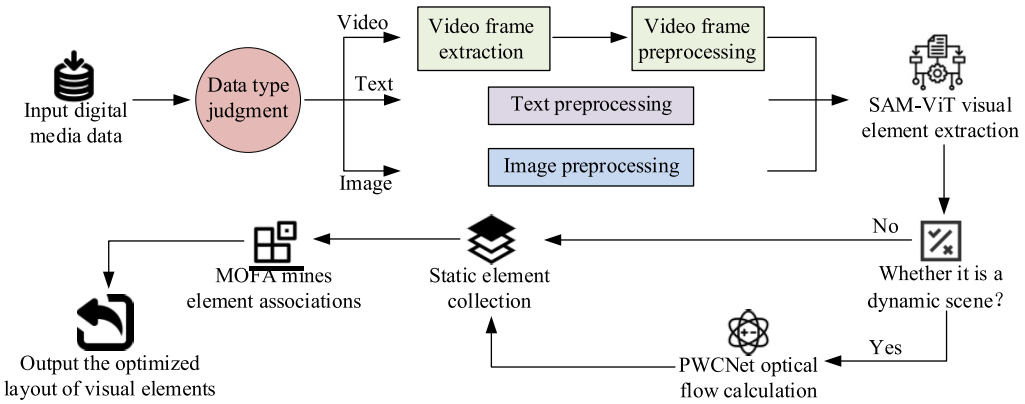


Fig. 6. Structural framework diagram of M-SVP model.

MOFA iteratively optimizes the multi-objective functions mentioned above to generate a Pareto optimal solution set, providing designers with diverse layout scheme options. The new visual element extraction and layout optimization model, which integrates SAM-ViT, PWCNet, and MOFA, is named M-SVP, and its structure is shown in Figure 6. The M-SVP model first classifies digital media visual elements into three types: images, texts, and videos. Then, it uses the SAM-ViT module for visual element segmentation and semantic classification to accurately extract static elements from images, texts, and videos. The PWCNet algorithm is applied to analyze the optical flow field in videos, capturing the motion trajectories of dynamic elements and supplementing spatiotemporal features. Finally, MOFA is used to mine the potential associations of multi-source data and generate layout constraint conditions. The layout optimization module combines aesthetic rules with spatial constraints, dynamically adjusting element positions and sizes through intelligent algorithms to support cross-device resolution adaptation. The model also ensures privacy protection by utilizing noise injection on edge devices and encrypted transmission for data security. It is suitable for scenarios such as advertisement design, e-commerce content generation, and AR interaction, significantly improving the semantic understanding accuracy and generation efficiency of visual layouts, while balancing functionality and security requirements. Among them, SAM-ViT achieves the precise extraction of visual elements through pixel-level segmentation. Its core is to minimize the segmentation error through the mask loss function:

$$L_{\text{seg}} = -\sum_{i=1}^{N} [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)] + \lambda \cdot \text{IoU}(M, M^*) , \tag{15}$$

where $y_i$ represents the true label of the pixel, $\hat{y}_i$ is the prediction probability of SAM-ViT, $M$ and $M^*$ are the element masks generated by the model, $\lambda$ is the weight coefficient, balancing the classification loss and mask accuracy. Compared with the traditional segmentation model, SAM-ViT directly optimizes the mask overlap degree by introducing the IoU term, making the convergence value of Equation (15) 15–20% lower than that of the comparison model, indicating a smaller segmentation error.

The MOFA algorithm transforms layout optimization into multi-objective function optimization, and the comprehensive objective is defined as

$$\min_{\Phi} \left[\omega_1 \cdot (1 - S) + \omega_2 \cdot (1 - A) + \omega_3 \cdot O + \omega_4 \cdot D\right] , \tag{16}$$

where $\Phi$ is the layout parameter, $S$ is the space occupancy rate, $A$ is the aesthetic score, $O$ is the element overlap rate, and $D$ is the element dispersion degree, $\omega_i$ representing different weight coefficients. MOFA achieves global optimization by simulating the luminous intensity of firefly populations. During the iterative process, the target value in Equation (16) is 12–18% lower than that of the genetic algorithm, and the convergence speed increases by 40%. Ultimately, it achieves a balanced layout with high space occupancy, high aesthetic score, and low overlap.

## 4. Verification of the Effects of the Improved SAM-ViT and MOFA Algorithms

### 4.1. Effectiveness verification of the improved SAM-ViT algorithm

In order to verify the performance superiority of the SAM-ViT and MOFA algorithms, the study compared it with three traditional object detection algorithm: YOLOv8, Mask Region-based Convolutional Neural Network (Mask R-CNN), and Residual Network-50 layers (ResNet-50). The experiments were conducted on an Ubuntu 20.04 LTS operating system with the PyTorch 2.0 deep learning framework, using Python 3.9 for programming. The hardware used included an NVIDIA GeForce RTX 3090 GPU, 128 GB of memory, and an Intel i9-12900K CPU. To ensure the reliability of the experiment, the PubLayNet [7, 31] and Magazine Layouts [9, 30] datasets were adopted. The two types of datasets combined cover more than 150 000 samples. Their annotation information directly corresponds to the full-process optimization goals of extraction, layout, and aesthetics of digital media visual elements, providing professional data support for the validity of the experiment. To ensure the reproducibility of the experiment, the SAM-ViT module optimizer adopts AdamW, the initial learning rate was set to $1 \times 10^{-4}$, and the batch size was 32. The stop criterion was that there were no improvement in the mean Intersection over Union (mIoU) of the validation set for 10 consecutive rounds or the number of training rounds reached 100. The PWCNet module optimizer was

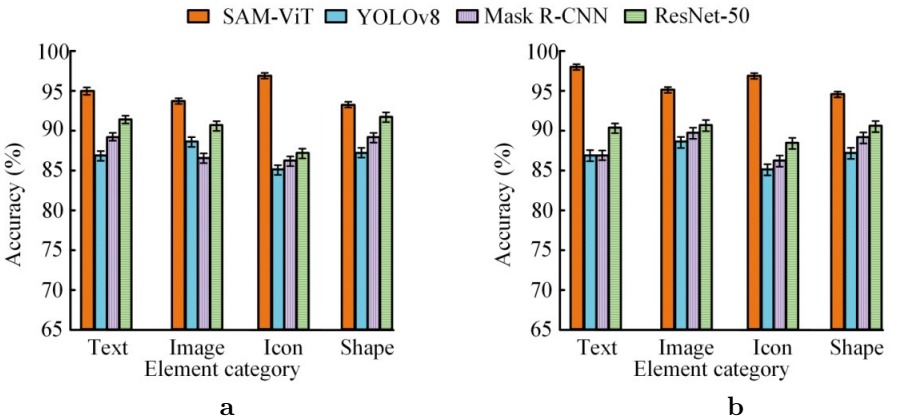


Fig. 7. Comparison of experimental results for accuracy in two datasets: (**a**) PubLayNet dataset; (**b**) Magazine Layouts dataset.

SGD, with an initial learning rate of 0.001 and a batch size of 16. The stopping criterion was to verify that the optical flow error of the collection has not decreased for eight consecutive rounds. The population size of the MOFA module was set to 50, the random step size factor $\alpha$ was 0.5, and the light intensity attenuation coefficient $\gamma$ was 0.2. The stop criterion was that the number of iterations reached 100 rounds or the multi-objective function value fluctuated less than $1 \times 10^{-3}$ for 15 consecutive rounds. SAM-ViT, YOLOv8, Mask R-CNN, and ResNet-50 were trained and tested on the two datasets for multi-class segmentation accuracy. The results are shown in Figure 7.

When trained on the PubLayNet dataset, SAM-ViT achieved a segmentation accuracy of $95.2 \pm 0.4\%$ for the text category, $94.6 \pm 0.3\%$ for the image category, $97.3 \pm 0.4\%$ for the icon category, and $93.3 \pm 0.2\%$ for the shape category. YOLOv8 achieved segmentation accuracy of $87.5 \pm 0.6\%$ for the text category and $85.3 \pm 0.8\%$ for the icon category. Mask R-CNN and ResNet-50 also showed lower accuracy in each element category compared to SAM-ViT. As shown in Figure 7b, when trained on the Magazine Layouts dataset, SAM-ViT achieved a segmentation accuracy of $98.8 \pm 0.2\%$ for the text category, $98.7 \pm 0.3\%$ for the icon category, and $95.9 \pm 0.2\%$ for the shape category. The accuracy of the other algorithms was significantly lower.

In conclusion, SAM-ViT demonstrated a clear accuracy advantage in classifying element categories across both datasets, outperforming the compared algorithms. The accuracy advantage of SAM-ViT stems from its integration of SAM's prompt-based segmentation mechanism and ViT's global semantic modeling capability. The former precisely locates the boundaries of elements, while the latter captures fine-grained features, effectively addressing the issues of ambiguous classification of small-sized elements

Tab. 1. Experimental results of robustness in complex environments.

| Experimental scene | Interference intensity | Evaluation index | SAM-ViT | YOLOv8 | Mask R-CNN | ResNet-50 |
|---|---|---|---|---|---|---|
| No interference | Accuracy rate [%] | - | $95.2 \pm 0.3$ | $87.5 \pm 0.4$ | $89.2 \pm 0.6$ | $91.7 \pm 0.6$ |
| Noise interference | Accuracy rate [%] | Gaussian noise (variance 0.01–0.05) | $92.3 \pm 0.3$ | $79.3 \pm 0.4$ | $81.6 \pm 0.6$ | $83.5 \pm 0.6$ |
| | Decline[%] | | 2.9 | 8.2 | 7.6 | 8.2 |
| Illumination variation | Accuracy rate [%] | Brightness ±30%, contrast ±20% | $91.7 \pm 0.3$ | $78.9 \pm 0.4$ | $80.9 \pm 0.6$ | $82.9 \pm 0.6$ |
| | Decline[%] | | 3.5 | 8.6 | 8.3 | 8.8 |
| Element occlusion | Accuracy rate [%] | Randomly block by 10% to 30% | $90.5 \pm 0.3$ | $77.8 \pm 0.4$ | $79.5 \pm 0.6$ | $81.3 \pm 0.6$ |
| | Decline[%] | | 4.7 | 9.7 | 9.7 | 10.4 |

and insufficient segmentation of complex backgrounds in contrast models. Therefore, it performs better.

To verify the robustness of SAM-ViT in complex environments, the study simulated three typical interference scenarios on the PubLayNet and Magazine Layouts datasets: (1) noise interference (adding Gaussian noise, variance 0.01–0.05); (2) lighting changes (adjust image brightness by ±30% and contrast by ±20%); (3) element occlusion (randomly occlusion 10% – 30% of the visual element area). The experimental results for consolidated datasets PubLayNet and Magazine Layouts are shown in Table 1.

To further investigate the segmentation accuracy of the SAM-ViT algorithm, the study evaluated the element masks and mIoU values of the four algorithms on two datasets: CIFAR-10 [10,11] and ISLVRC2012 [3,4,20]. The evaluation results are shown in Figure 8. As shown in Figure 8a, on the CIFAR-10 dataset, SAM-ViT achieved an initial mIoU of 0.75 after 10 training epochs. As the training progressed, it rapidly increased to 0.95 after 50 epochs and stabilized at 0.95. In comparison, YOLOv8 started with an mIoU of about 0.73 and reached 0.92 at the end. As shown in Figure 8b, on the ISLVRC2012 dataset, SAM-ViT's segmentation accuracy showed only slight fluctuations and ultimately stabilized at 0.95. YOLOv8 started at approximately 0.73 and stabilized at 0.92. Mask R-CNN stabilized at 0.91, and ResNet-50 stabilized at around 0.87. In summary, on both datasets, SAM-ViT consistently outperformed other algorithms in mIoU, achieving higher values more quickly and maintaining stability, highlighting its superior performance in element segmentation accuracy. The dynamic mask generation ability of SAM can accurately depict the boundaries of elements and reduce segmentation deviations. The self-attention mechanism of ViT can efficiently learn global feature associations and accelerate model convergence. The combination of the two enables it

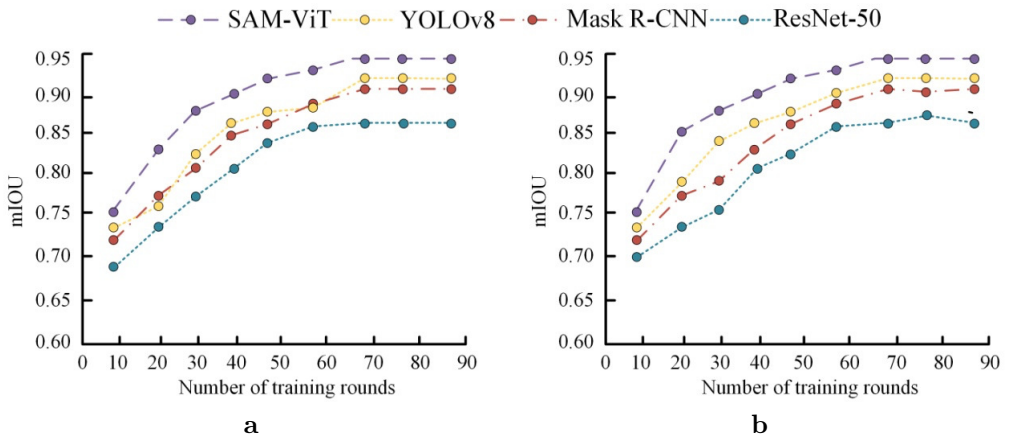


Fig. 8. Comparison of experimental results for mIoU in two datasets: (**a**) CIFAR-10; (**b**) ISLVRC 2012.

Tab. 2. Experimental results of precision, predicted recall, and F1 score.

| Dataset | Algorithm | Precision [%] | Recall [%] | F1 score [%] |
|---|---|---|---|---|
| ISLVRC2012 | SAM-ViT | $98.22 \pm 0.35$ | $97.89 \pm 0.41$ | $98.17 \pm 0.38$ |
| | YOLOv8 | $93.12 \pm 0.52$ | $90.62 \pm 0.58$ | $91.35 \pm 0.55$ |
| | Mask R-CNN | $89.26 \pm 0.61$ | $90.25 \pm 0.65$ | $92.48 \pm 0.63$ |
| | ResNet-50 | $84.75 \pm 0.73$ | $89.68 \pm 0.78$ | $88.12 \pm 0.75$ |
| CIFAR-10 | SAM-ViT | $97.78 \pm 0.39$ | $98.56 \pm 0.43$ | $97.74 \pm 0.40$ |
| | YOLOv8 | $92.54 \pm 0.56$ | $90.66 \pm 0.61$ | $89.46 \pm 0.59$ |
| | Mask R-CNN | $90.11 \pm 0.64$ | $88.95 \pm 0.69$ | $92.13 \pm 0.66$ |
| | ResNet-50 | $83.97 \pm 0.76$ | $87.96 \pm 0.82$ | $90.17 \pm 0.79$ |

to achieve high-bit accuracy more quickly during training and maintain stability, which is superior to the compared algorithms.

To further showcase the performance of the SAM-ViT algorithm, the study compared the four algorithms based on precision, predicted recall, and F1 score. The comparison results are shown in Table 2. In this Table it can be seen that when tested on the ISLVRC2012 dataset, SAM-ViT achieved a precision of $98.22 \pm 0.35$, predicted recall of $97.89 \pm 0.41$, and F1 score of $98.17 \pm 0.38$. YOLOv8's precision and predicted recall were $93.12 \pm 0.52\%$ and $90.62 \pm 0.58\%$, respectively, with an F1 score of $91.35 \pm 0.55\%$. SAM-ViT showed advantages in all three metrics. When tested on the CIFAR-10 dataset, Mask R-CNN's predicted recall and F1 score were $88.95 \pm 0.69\%$ and $92.13 \pm 0.66\%$, respectively, both lower than SAM-ViT's values. In both datasets, ResNet-50's metrics did not exceed 90%. In conclusion, SAM-ViT's intelligent extraction and segmentation algorithm outperforms other mainstream algorithms in terms of performance.

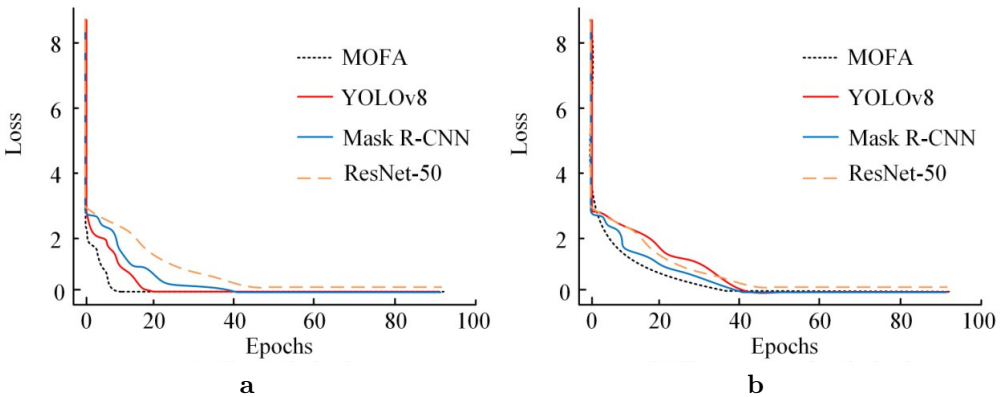


Fig. 9. Loss rate convergence results. (**a**) Optimization objective is less than 10. (**b**) Number of optimization targets is greater than 10.

To verify whether the MOFA algorithm can maintain high performance when dealing with data of different scales, the study further compared the loss rate convergence of the four algorithms, as shown in Figure 9. As shown in Figure 9a, in the scenario where the number of optimization objectives is less than 10, when MOFA is trained to 15 rounds, the MOFA loss drops to 0.52, which is significantly ahead of the convergence speed of YOLOv8, Mask R-CNN, and ResNet-50, demonstrating the global optimization efficiency of swarm intelligence algorithms in low-dimensional objectives. However, as shown in Figure 9b, when the number of optimization objectives is greater than 10, the MOFA loss value remains at 0.63 after 40 rounds of training. Compared with its own low-dimensional scenario, the number of iterations for MOFA to converge to the same loss increases from 18 rounds to 42 rounds, revealing that when dealing with complex multi-objective optimization problems, the Firefly algorithm is prone to falling into local optima. This leads to a significant decline in both convergence efficiency and stability.

### 4.2. Evaluation of the intelligent extraction and layout model based on SAM-ViT and MOFA

After verifying the superiority of SAM-ViT, the study further analyzed the performance of the intelligent extraction and layout model M-SVP, which integrates SAM-ViT and MOFA, by comparing it with models built using YOLOv8 combined with Genetic Algorithm (GA-YOLOv8), Mask R-CNN, and ResNet-50. The experiments were conducted with PyTorch as the core deep learning framework, based on the Anaconda 3 development environment, and training was performed in the MATLAB R2023b simulation environment. To determine whether M-SVP can accurately achieve visual element extraction and layout optimization, the research focuses on the core pain point of the

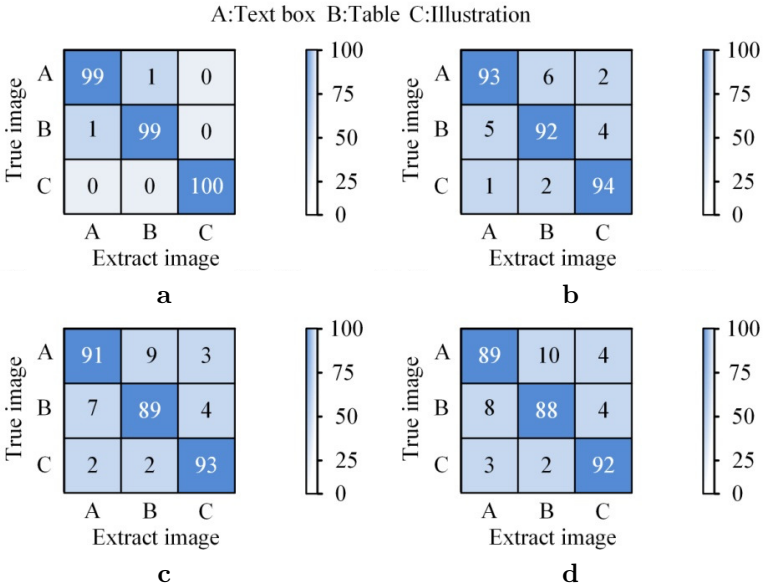


Fig. 10. Comparison of recognition and confusion of visual elements in models: (**a**) M-SVP; (**b**) GA-YOLOv8; (**c**) Mask R-CNN; (**d**) ResNet-50.

layout task in the PubLayNet and Magazine Layouts datasets – misjudgment of confusing elements can directly lead to layout logic disorder. Therefore, three types of typical confusing elements in the two datasets are selected, and 100 samples are taken for each. The extraction and discrimination capabilities of the four algorithms for these highly similar elements were compared, and the results are shown in Figure 10.

By observing the confusion matrix in Figure 10, it is found that the M-SVP model misjudges one of the 100 text box samples as a table and one of the 100 table samples as a text box. The extraction of the remaining samples is all correct, with an accuracy rate of over 99%. However, GA-YOLOv8, Mask R-CNN and ResNet-50 are more prone to confusion in distinguishing between text boxes and tables. For example, ResNet-50 misjudged 10 out of 100 text box samples as tables. To sum up, the M-SVP model has a higher accuracy extraction rate for easily confused visual elements (such as layout elements with similar structures like text boxes and tables), and it has a prominent advantage in the accuracy of visual element recognition, significantly outperforming other models. The high extraction rate of confusable elements by M-SVP is attributed to the precise capture of fine-grained features by its SAM-ViT module. Combined with the enhanced attention mechanism of the model for confusable category features, it

effectively reduces the misjudgment caused by the interference of similar features, thus performing better.

To further verify the adaptability of the model in cross-cultural scenarios, the study selected typical visual samples from three cultural backgrounds: the East, the West, and the Middle East. The aesthetic score performance of M-SVP and the contrast model under different cultural aesthetic standards was compared. To ensure the objectivity and reliability of aesthetic scoring, the experiment recruited 10 professional raters and 30 ordinary users as the scoring subjects, all of whom scored the content anonymously and independently.

**Scoring protocol**

Based on the internationally recognized visual aesthetics assessment framework, a scale of 1 to 100 points was adopted. Each rater scored the same sample twice, and the average of the two scores was taken as the individual scoring result. Then, the average of all raters was calculated as the final aesthetic score.

**Consistency test among raters**

Consistency was verified by the intraclass correlation coefficient (ICC). The ICC for professional raters was $0.89 (p < 0.001)$, and that for ordinary users was $0.82$ $(p < 0.001)$, both of which were higher than the reliable threshold of 0.7, indicating stable scoring results.

**Statistical significance**

One-way ANOVA was conducted on the aesthetic scores of different models, and the results showed that the differences between the models were statistically significant. The post hoc Tukey HSD test further indicated that the score differences between the M-SVP and the control models reached a significant level ($p < 0.01$), confirming that the aesthetic optimization effect was not a random error.

These results are shown in Figure 11. It can be seen from Figure 11a that after the layout optimization of the M-SVP model under the background of Eastern culture, the aesthetic score of the sample has increased to 95.6 points, showing a considerable degree of optimization. After layout optimization of the GA-YOLOv8 model, the aesthetic score increased to 78.5 points, which was lower than the optimization degree of the M-SVP model. It can be seen from Figure 11b that in the context of Western culture, after the optimization of the M-SVP model, the aesthetic score of the sample increased to 97.8 points, while the aesthetic score of the corresponding sample of the GA-YOLOv8 model decreased to 75.6 points. The aesthetic scores of the visual elements of the other two comparison models were not significantly optimized. To sum up, the M-SVP model shows good adaptability when facing users from different cultural backgrounds, and it can significantly improve user experience and effectively convey information.

To further assess the model's robustness, the study compared the fluctuation in overall layout quality after adding $\pm 10\%$ size scaling and $\pm 5\%$ position shift disturbances to

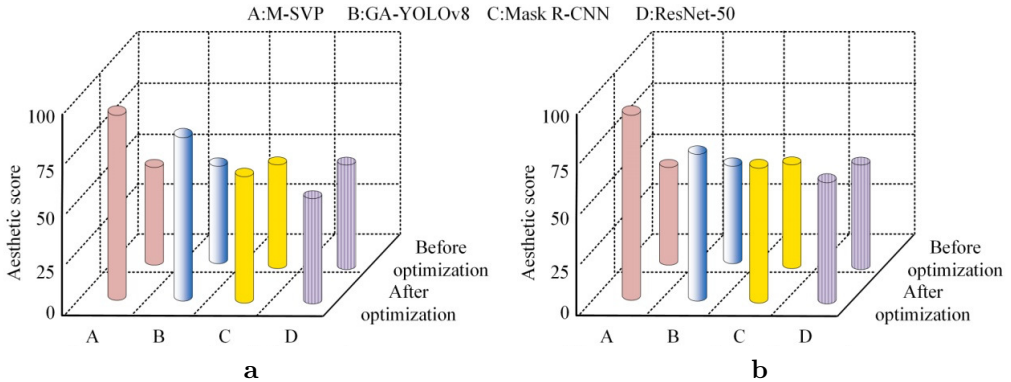


Fig. 11. Comparison of aesthetic scores in the context of different cultural backgrounds. (**a**) Eastern culture; (**b**) Western culture.

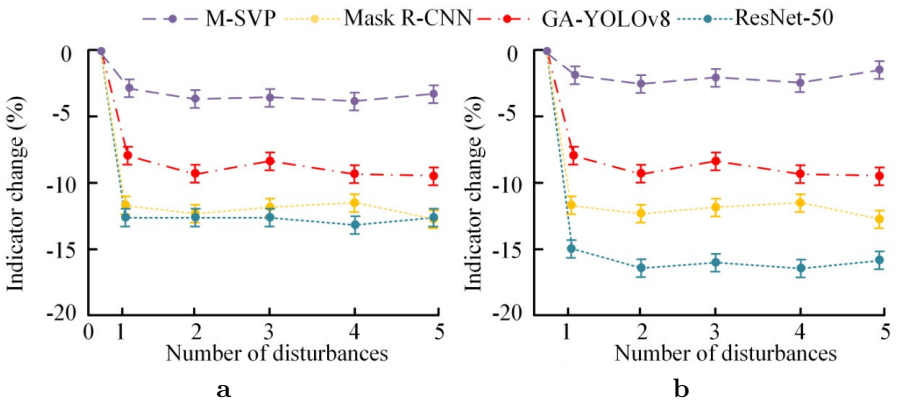


Fig. 12. Comparison of changes in two measures after disturbance: (**a**) in aesthetic scores; (**b**) in space occupancy.

the four models. The results are shown in Figure 12. After five disturbances, the M-SVP model showed minimal fluctuation in aesthetic scores and spatial occupancy rates, with changes of −3.2% and −2.3%, respectively. The GA-YOLOv8 model's robustness was second only to M-SVP, with fluctuations of around 10%. The other two models showed weaker robustness, with changes in both metrics exceeding 10% after five disturbances.

In conclusion, the M-SVP model demonstrated significant advantages in layout robustness, making it suitable for applications such as academic papers and technical reports while providing visual support for the model's practicality. The layout robustness advantage of M-SVP stems from the global optimization and dynamic adjustment

Tab. 3. Progressive verification of modules in the ablation study.

| Model Variants | Aesthetic score [0, 100] | mIoU (segmentation accuracy) [0, 1] | Space utilization rate [%] | Inference time [ms] |
|---|---|---|---|---|
| Manual rules + traditional CNN | 60.2 | 0.60 | 65.0 | 30.5 |
| ViT | 68.5 | 0.65 | 68.3 | 40.2 |
| SAM-ViT | 75.8 | 0.80 | 72.5 | 50.3 |
| SAM-ViT-PWCNet | 82.1 | 0.83 | 78.6 | 60.5 |
| M-SVP | 97.8 | 0.95 | 85.2 | 70.1 |

capabilities of the MOFA algorithm, which can rapidly iterate and optimize the layout parameters when interference occurs. Combined with the real-time modeling of element association by PWCNet, the layout imbalance caused by interference can be effectively offset. However, the contrast model, lacking this dynamic adaptation mechanism, finds it difficult to handle fluctuations in element position or size caused by interference, and thus has relatively weak robustness. Furthermore, in order to clarify the collaborative gain of each core module in the M-SVP model and its impact on the computational cost, manual rules combined with the traditional CNN method were selected as the benchmark, along with basic ViT, SAM-ViT, SAM-ViT-PWCNet, and the complete M-SVP model.

Ablation experiments were conducted on aesthetic scores, mIoU, space occupancy rate and reasoning time indicators, and the results are shown in Table 3. In this Table it can be seen that the complete M-SVP model comprehensively outperforms other variants in core indicators. Its highest aesthetic score is 97.8, the highest space occupancy rate reaches 85.2%, and it maintains the same segmentation accuracy as SAM-ViT and SAM-VIT-PWCNet. This indicates that mIoU is mainly determined by the SAM-ViT module. It is worth noting that its reasoning time is the longest, which is due to the integration of the full modules of SAM-ViT, PWCNet and MOFA. Specifically, the combination of manual rules and traditional CNNS as the baseline model has the poorest performance, with an aesthetic score of only 60.2 and a space occupancy rate of 65.0%, highlighting the limitations of non-intelligent approaches. The basic ViT model has improved to some extent compared with the baseline, but it still lags significantly behind SAM-ViT. The mIoU of the latter was 23.1% higher than that of ViT, and the aesthetic score was 10.7% higher, verifying the key role of SAM enhancement in the precise extraction of visual elements. Further comparison shows that the space occupancy rate of the SAM-VIT-PwCNet variant is 8.4% higher than that of SAM-ViT. This is because the dynamic element correlation analysis of PWCNet reduces layout conflicts, but the inference time correspondingly increases by 10.2 ms. Ultimately, compared with SAM-ViT-PWCNet, the complete M-SVP model integrating MOFA has a further 8.8% improvement in aesthetic score and an 8.4% increase in space occupancy rate, but the reasoning time has increased by another 9.6 ms.

Tab. 4. The summary table of core performance indicators comparison on the supplementary dataset.

| Test Dataset | Evaluation Metric | M-SVP | GA-YOLOv8 | Mask R-CNN | ResNet-50 |
|---|---|---|---|---|---|
| E-commerce Product Detail Page | Extraction Accuracy [%] | 91.5 | 78.3 | 76.9 | 72.1 |
| | mIOU | 0.86 | 0.71 | 0.69 | 0.63 |
| | Aesthetic Score (0–100) | 85.2 | 68.7 | 65.3 | 60.5 |
| | Space Utilization Rate [%] | 80.3 | 65.2 | 63.7 | 59.8 |
| Social Media Post | Extraction Accuracy [%] | 89.7 | 75.6 | 73.2 | 68.9 |
| | mIOU | 0.84 | 0.68 | 0.65 | 0.59 |
| | Aesthetic Score (0–100) | 82.6 | 66.3 | 62.8 | 58.2 |
| | Space Utilization Rate [%] | 78.5 | 63.1 | 60.5 | 57.3 |

In summary, the progressive performance of each variant confirms the collaborative value of the modules. SAM enhances the segmentation accuracy, PWCNet optimizes the efficiency of dynamic layout, and MOFA strengthens the aesthetic and spatial presentation. Although the complete M-SVP model has the best comprehensive performance, its inference time is nearly double that of the baseline model, reflecting the computational cost of integrated multi-module intelligence and highlighting the trade-off between performance gain and computational overhead.

To verify the generalization ability of the M-SVP model in unseen digital media scenarios, two types of external datasets with significant differences from the training set scenarios were selected: the E-commerce Product Detail Page dataset [26], containing 50 000 samples, covering pages of clothing, electronics, and food, characterized by a dense arrangement of *multiple images + short text + price tags*, and the Social Media Post dynamic dataset [25], containing 30 000 samples, covering WeChat official accounts and Weibo images and text, characterized by *irregular layout + mixed emoticons/topic tags*). On the above datasets, the core metrics of M-SVP were compared with those of GA-YOLOv8, Mask R-CNN, and ResNet-50, and the results are shown in Table 4. These results indicate that M-SVP still maintains high performance in two types of unfamiliar scenarios: the extraction accuracy rate exceeds 89% in both cases, mIOU is $\geq 0.84$, the aesthetic score and space occupancy rate only decrease by 5–8% compared with the training set, while the performance degradation of the comparison models generally reaches 15–25%. In conclusion, M-SVP demonstrates strong generalization ability in cross-scenario tasks, verifying its practicality as a general digital media processing solution.

To visually present the focus of attention and correlation logic of the model layout decision, the typical digital media scenario of the social media page is still selected. The specific layout decision process of M-SVP is analyzed through the attention weight graph, and the result is shown in Figure 12. This Figure shows the attention weight

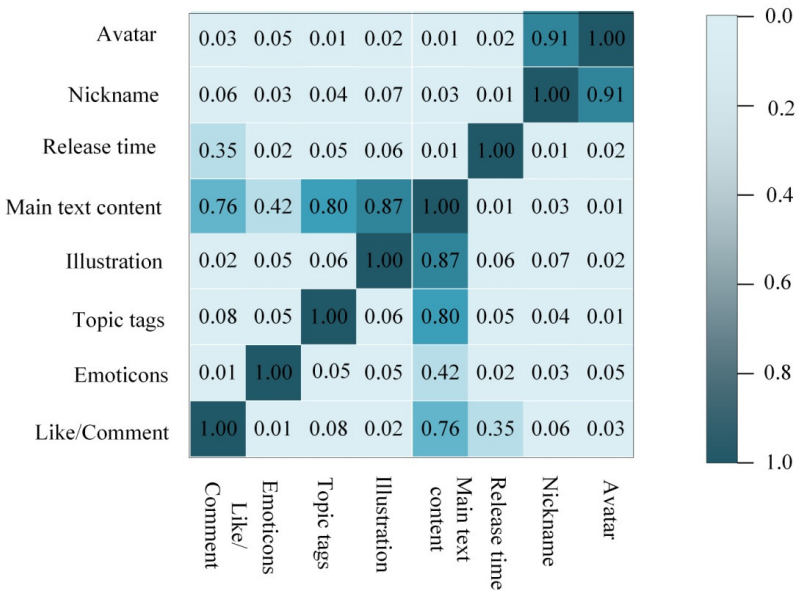


Fig. 13. Social media page model layout attention weight chart.

heat map of eight core elements in social media pages. The correlation strength between elements is quantified through color depth and numerical values. Among them, the correlation degree between avatars and nicknames exceeds 0.9, and a strong binding of *visual identity* + *text identity* is used to build a fast recognition channel for users about the identity of the publisher. The main text is deeply associated with the accompanying images (0.87) and topic tags (0.80), prioritizing the strengthening of the collaborative relationship between *text information – visual supplementation – dissemination classification* to meet the dissemination needs of *efficient content reach* on social media. The correlation between the main text and the interactive area (0.76) is particularly significant. Through the connection of *core content* → *interactive feedback*, the willingness to interact is strengthened, and it precisely alters the scene behavior pattern of *emotion-driven interaction*. The overall weight distribution clearly echoes the scene function chain of *identity recognition – content understanding – interactive dissemination*, intuitively verifying the scene pertinence and logical rationality of the model layout decision.

## 5. Conclusion

To address the issues of fuzzy visual element extraction and layout optimization, which rely on manual experience and lead to a balance problem between efficiency and aesthetics in digital media, the study designed the M-SVP model. This model constructs a multimodal collaborative architecture, covering core modules for precise visual element extraction, dynamic correlation analysis, and intelligent layout optimization, offering an intelligent solution for automated digital media design. The experimental results showed that the SAM-ViT algorithm achieved the highest visual element extraction accuracy of 98.8% across different categories. As the number of training iterations increased to 50, its mIoU value stabilized at 0.95, and its F1 score reached a maximum of $98.17 \pm 0.38\%$. Furthermore, the M-SVP model demonstrated 99% extraction accuracy for easily confused visual elements. After layout optimization, the M-SVP model's aesthetic score improved to 95.6, and its spatial occupancy rate increased to 97.2%, far exceeding the comparison models. In conclusion, the M-SVP model exhibited excellent performance in information extraction, layout optimization, and robustness testing. However, the model itself still has certain limitations. Its multi-module integration leads to high computational complexity and insufficient real-time performance when deployed on edge devices with limited computing power. Future work will focus on optimizing the above-mentioned deficiencies, compressing model parameters through knowledge distillation to reduce computational costs, and further enhancing the practicality and universality of the model.

## Fundings

The research is supported by Shanxi Provincial Education Science *14th Five-Year Plan* 2022 annual general planning project *New Era of higher vocational colleges* Computer Basics *course ideological and political implementation path* (project number: GH-220519); Shanxi Engineering Vocational College 2022 annual teaching and research projects *Research on University Computer Basic Teaching Based on Computational Thinking* (Project number: JY2022-12).

## References

[1] J. D. Blair, K. M. Gaynor, M. S. Palmer, and K. E. Marshall. A gentle introduction to computer vision-based specimen classification in ecological datasets. *Journal of Animal Ecology* 93(2):147–158, 2024. doi:10.1111/1365-2656.14042.

[2] F. Chen, L. Chen, H. Han, S. Zhang, D. Zhang, et al. The ability of Segmenting Anything Model (SAM) to segment ultrasound images. *BioScience Trends* 17(3):211–218, 2023. doi:10.5582/bst.2023.01128.

[3] J. Deng, A. Berg, S. Satheesh, H. Su, A. Khosla, et al. ImageNet Large Scale Visual Recognition Challenge 2012 (ILSVRC2012). IMAGENET, 2012. https://www.image-net.org/challenges/LSVRC/2012/.

[4] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, et al. ImageNet: A large-scale hierarchical image database. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 248–255, 2009. doi:10.1109/CVPR.2009.5206848.

[5] J.-H. Ha and H. Lee. A deep learning model for precipitation nowcasting using multiple optical flow algorithms. *Weather and Forecasting* 39(1):41–53, 2024. doi:10.1175/WAF-D-23-0104.1.

[6] E. Hassan, M. Y. Shams, N. A. Hikal, and S. Elmougy. The effect of choosing optimizer algorithms to improve computer vision tasks: A comparative study. *Multimedia Tools and Applications* 82(11):16591–16633, 2023. doi:10.1007/s11042-022-13820-0.

[7] A. J. Jimeno Yepes. PubLayNet. GitHub, 2025. https://github.com/ibm-aur-nlp/PubLayNet.

[8] S. Khoubani and M. H. Moradi. A deep learning phase-based solution in 2D echocardiography motion estimation. *Physical and Engineering Sciences in Medicine* 47(4):1691–1703, 2024. doi:10.1007/s13246-024-01481-2.

[9] S. Kitada. huggingface-datasets_Magazine. GitHub, 2023. https://github.com/creative-graphic-design/huggingface-datasets_Magazine.

[10] A. Krizhevsky. *Learning Multiple Layers of Features from Tiny Images*. Master's thesis, University of Toronto, 2009. https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

[11] A. Krizhevsky, V. Nair, and G. Hinton. The CIFAR-10 dataset. In Alex Krizhevsky home page, 2009. https://www.cs.toronto.edu/~kriz/cifar.html.

[12] M. Y. Landolsi, L. Hlaoua, and L. Ben Romdhane. Information extraction from electronic medical documents: state of the art and future research directions. *Knowledge and Information Systems* 65(2):463–516, 2023. doi:10.1007/s10115-022-01779-1.

[13] C. Li, Y. Huang, W. Li, H. Liu, X. Liu, et al. Flaws can be applause: Unleashing potential of segmenting ambiguous objects in SAM. *Advances in Neural Information Processing Systems* 37:45578–45599, 2024. doi:10.52202/079017-1449.

[14] J. Li, Z. Zhou, J. Yang, A. Pepe, C. Gsaxner, et al. MedShapeNet – a large-scale dataset of 3D medical shapes for computer vision. *Biomedical Engineering / Biomedizinische Technik* 70(1):71–90, 2025. doi:10.1515/bmt-2024-0396.

[15] H. B. Mahajan, N. Uke, P. Pise, M. Shahade, V. G. Dixit, et al. Automatic robot manoeuvres detection using computer vision and deep learning techniques: a perspective of internet of robotics things (IoRT). *Multimedia Tools and Applications* 82(15):23251–23276, 2023. doi:10.1007/s11042-022-14253-5.

[16] T. Onyejelem and A. Eric Msughter. Digital generative multimedia tool theory (DGMTT): A theoretical postulation. *Journalism and Mass Communication* 14(3):189–204, 2024. doi:10.17265/2160-6579/2024.03.004.

[17] A. S. Ortega-Calvo, R. Morcillo-Jimenez, C. Fernandez-Basso, K. Gutiérrez-Batista, M. A. Vila, et al. AIMDP: An artificial intelligence modern data platform. Use case for Spanish national health service data silo. *Future Generation Computer Systems* 143:248–264, 2023. doi:10.1016/j.future.2023.02.002.

[18] E. W. Prastyaningtyas, A. M. A. Ausat, L. F. Muhamad, M. I. Wanof, and S. Suherlan. The role of information technology in improving human resources career development. *Jurnal Teknologi Dan Sistem Informasi Bisnis* 5(3):266–275, 2023. doi:10.47233/jteksis.v5i3.870.

[19] B. Rokh, H. Mirvaziri, and M. H. Olyaee. A new evolutionary optimization based on multi-objective firefly algorithm for mining numerical association rules. *Soft Computing* 28(9):6879–6892, 2024. doi:10.1007/s00500-023-09558-y.

[20] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, et al. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision* 115(3):211–252, 2015. doi:10.1007/s11263-015-0816-y.

[21] S. R. Shen, J. Balakrishnan, and C. H. Cheng. Dynamic content layout optimization for news website front pages. *Journal of Modelling in Management* 19(6):1907–1926, 2024. doi:10.1108/JM2-01-2024-0015.

[22] K. Subramanian, F. Hajamohideen, V. Viswan, N. Shaffi, and M. Mahmud. Exploring intervention techniques for Alzheimer's disease: Conventional methods and the role of AI in advancing care. *Artificial Intelligence and Applications* 2(2):59–77, 2024. doi:10.47852/bonview42022497.

[23] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8934–8943, 2018. doi:10.1109/CVPR.2018.00931.

[24] K. Tan, J. Wu, H. Zhou, Y. Wang, and J. Chen. Integrating advanced computer vision and AI algorithms for autonomous driving systems. *Journal of Theory and Practice of Engineering Science* 4(1):41–48, 2024. doi:10.53469/jtpes.2024.04(01).06. https://centuryscipub.com/index.php/jtpes/article/view/427.

[25] P. Tank. Social media post dataset. Kaggle Dataset, 2024. https://www.kaggle.com/datasets/prishatank/post-generator-dataset/data.

[26] F. Tiago. ecommerce-product-dataset. GitHub Repository, 2025. https://github.com/octaprice/ecommerce-product-dataset.

[27] D. Wang, J. Zhang, B. Du, M. Xu, L. Liu, et al. SAMRS: Scaling-up remote sensing segmentation dataset with segment anything model. *Advances in Neural Information Processing Systems* 36:8815–8827, 2023. doi:10.48550/arXiv.2305.02034.

[28] Y. Xue and J. Williams. Inducing shifts in attentional and preattentive visual processing through brief training on novel grammatical morphemes: An event-related potential study. *Language Learning* 74(S1):185–223, 2024. doi:10.1111/lang.12642.

[29] P. Zhang, J. Zheng, H. Lin, C. Liu, Z. Zhao, et al. Vehicle trajectory data mining for artificial intelligence and real-time traffic information extraction. *IEEE Transactions on Intelligent Transportation Systems* 24(11):13088–13098, 2023. doi:10.1109/TITS.2022.3178182.

[30] X. Zheng, X. Qiao, Y. Cao, and R. W. Lau. Content-aware generative modeling of graphic design layouts. *ACM Transactions on Graphics* 38(4):133, 2019. doi:10.1145/3306346.3322971.

[31] X. Zhong, J. Tang, and A. Jimeno Yepes. PubLayNet: Largest dataset ever for document layout analysis. In: *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1015–1022, 2019. doi:10.1109/ICDAR.2019.00166.

# Deep Learning for Semantic Segmentation of Linear Infrastructure from UAV Imagery Using NVIDIA Jetson AGX Orin

Justyna S. Stypułkowska*
*Łukasiewicz Research Network – Institute of Aviation, Warsaw, Poland*
**Corresponding author: Justyna S. Stypułkowska (justyna.stypulkowska@ilot.lukasiewicz.gov.pl)*

**Submitted:** 30 Aug 2025 **Accepted:** 04 Dec 2025 **Published:** 31 Mar 2026 

**Abstract** A method for semantic segmentation of RGB images captured by UAVs to detect railway infrastructure elements, including tracks, level crossings, and surrounding vegetation is proposed. The study was conducted at the Łukasiewicz Research Network – Institute of Aviation, where a proprietary, manually annotated UAV RGB dataset was created. Five deep neural network architectures were trained and compared: DeepLabV3+, Feature Pyramid Network (FPN), LinkNet, Pyramid Attention Network (PAN) and X-Unet. These models were chosen for their distinct approaches to semantic segmentation and feature processing. Training was performed on a desktop computer with an NVIDIA GeForce RTX 3080 GPU and tests were made also on an NVIDIA Jetson AGX Orin to assess deployment feasibility under real-time conditions. Experimental results confirm the strong performance of the analyzed models in segmenting railway tracks and surrounding vegetation. FPN achieved the highest scores, followed by X-Unet, DeepLabV3+, LinkNet, and PAN. All models operated reliably on the NVIDIA Jetson AGX Orin edge platform. The proposed solution can support remote monitoring of railway infrastructure and vegetation. It can also be adapted to other applications by adjusting the training dataset and object categories. This research demonstrates the potential of deep learning as a powerful tool for analyzing UAV RGB imagery in engineering and environmental contexts.

**Keywords:** artificial intelligence, deep learning, semantic segmentation, RGB imaging, UAV, NVIDIA Jetson AGX Orin, railway infrastructure, image analysis, real-time monitoring, DeepLabV3+, FPN, PAN, LinkNet, X-Unet.

## 1. Introduction

Automation of railway infrastructure monitoring can play a significant role in ensuring safety, optimizing maintenance, and controlling vegetation encroaching on the railway right-of-way. Traditional manual inspection methods, although effective, are time-consuming, costly, and prone to subjective errors. These limitations become even more critical in the context of increasing railway traffic and the need for predictive maintenance strategies [19]. Recent studies highlight that automation not only reduces operational costs but also improves reliability and consistency compared to manual inspections [18].

In response to these challenges, this study introduces a semantic segmentation approach for UAV-acquired RGB images using deep learning algorithms. The research aims to automate the detection of key objects (e.g., railway tracks, level crossings, vegetation) and compare the effectiveness of selected neural network architectures, while assessing their deployment compatibility on an NVIDIA Jetson AGX Orin mounted on

a UAV-based module. This dual focus, accuracy and embedded feasibility, addresses a critical gap in the literature, where most works concentrate on segmentation quality without considering deployment constraints on edge AI platforms [28].

The literature offers various methods for object segmentation and aerial infrastructure analysis. Few studies compare the performance of different deep learning models in this specific context; this research addresses that gap. The work presented was carried out as part of the statutory project of the Łukasiewicz Research Network – Institute of Aviation entitled *Assessment and analysis of the potential use of remote object detection methods and condition identification of critical infrastructure*. The originality of this work lies in its comprehensive evaluation of five architectures under identical conditions and practical deployment on an embedded platform, ensuring findings are both theoretically relevant and applicable in real-world UAV monitoring scenarios [11].

Five modern segmentation architectures were trained: DeepLabV3+ [5, 6, 7], Feature Pyramid Network (FPN) [12, 15, 16, 20], LinkNet [4], Pyramid Attention Network (PAN) [14], and X-Unet [21, 22, 25, 31]. All models were trained on the same training dataset to ensure a fair and reliable comparison. Such a standardized setup is rarely reported, as existing studies often use different datasets or protocols, making direct comparisons difficult [3].

The study also enabled an evaluation of the effectiveness of each architecture under real-world application conditions. Each model was assessed in terms of segmentation quality using the following metrics: Pixel Accuracy, IoU, mean IoU, F1-score, mean F1-score, Precision, and Recall. Additionally, the target model's response time was analyzed during deployment on the NVIDIA Jetson AGX Orin to assess real-time monitoring feasibility. This aspect is crucial for UAV-based systems, where latency affects operational safety and decision-making [1].

Evaluating segmentation quality helps to understand how accurately each model represents railway infrastructure objects, which are crucial for effective track condition monitoring and vegetation overgrowth assessment. Inference speed determines the practical feasibility of deploying the solution on a UAV with NVIDIA Jetson AGX Orin, where information must be delivered in real time. This study helps bridge the gap between algorithmic development and practical implementation in edge computing environments [30].

This dual evaluation approach provides both theoretical insights and practical validation of the proposed solutions. The results also lay the groundwork for future research on optimizing segmentation models for other linear infrastructure types and integrating multimodal data (e.g., LiDAR, thermal imaging) to enhance the monitoring [23].

The following sections examine which architectures achieve the highest segmentation accuracy. Identifying the most precise models enables transferring the solution to other infrastructure monitoring systems or extending it to new object classes.

## 2. Related works

There is a growing interest in methods that leverage deep neural networks for semantic segmentation of linear infrastructure imagery, including railway infrastructure. Of particular relevance is the use of images acquired from unmanned aerial vehicles (UAVs), which enable rapid and accurate data collection over relatively large and, in some cases, difficult-to-access areas. Recent studies emphasize that UAV-based monitoring significantly improves coverage and operational efficiency compared to traditional ground inspections [1, 19].

The literature describes various methods aimed at detecting railway tracks, safety barriers, and other environmental elements. Most existing approaches utilize U-Net architectures and their modifications for semantic segmentation tasks [22]. Alternatively, architectures such as DeepLabV3+ [7], PSPNet, and Feature Pyramid Networks (FPN) [15] are also employed. In addition, attention-based and transformer-enhanced variants of DeepLabV3+ have recently been proposed to improve feature extraction in complex aerial scenes [2, 28].

Some studies use DeepLabV3+ for railway track segmentation in UAV imagery, often paired with lightweight backbones like MobileNetV2 or MobileNetV3 to enable deployment on resource-constrained platforms. One enhanced DeepLabV3+ variant with MobileNetV3 achieved 97.69% accuracy and 88.93% mean IoU, while maintaining good computational efficiency for edge devices such as NVIDIA Jetson [27]. Similar improvements have been reported for adaptive DeepLab variants optimized for UAV imagery, achieving high IoU scores while reducing inference time [2].

Other researchers describe the use of semantic segmentation techniques for detecting elements such as railway tracks, sleepers, and components of catenary systems (OCS). Their work provides a broad overview of methods addressing both segmentation and damage detection of technical components, employing architectures such as FCN and U-Net, LiDAR data, as well as multimodal approaches. This highlights the growing importance of deep learning in this field [10]. Recent multimodal approaches integrate LiDAR and thermal imaging to enhance detection under adverse conditions, such as fog or low light [24, 26].

Despite the focus of many studies on improving segmentation accuracy, relatively few address the implementation of developed solutions on edge computing devices. This aspect is crucial for real-time operation, for example during UAV flights. Some recent publications have responded to this need, demonstrating that image segmentation models can be deployed on small, resource-constrained devices. One example introduces a lightweight U-Net variant tested on CPUs, GPUs, and FPGAs, achieving real-time performance, with the best results on FPGA-based deployment. Lopez-Montiel et al. [17] proposed JetSeg, a simplified segmentation model optimized for Jetson Xavier, offering lower resource consumption and faster operation than typical alternatives, making it

suitable for embedded systems. Further research confirms that hardware-aware designs and attention-based optimizations can significantly improve inference speed on Jetson platforms [13, 23].

Chen and colleagues [8] described a model that can be deployed on edge devices such as the Jetson Nano. Their system is capable of performing real-time image segmentation at approximately 25 frames per second, without the need to transmit data to a cloud environment. This means that railway track monitoring can be performed directly onboard the UAV. Similar results have been achieved using global–local attention mechanisms for UAV imagery, enabling real-time segmentation with reduced latency [30].

Despite these examples, few studies compare segmentation quality and edge performance across multiple architectures. This work addresses that gap by evaluating five architectures: DeepLabV3+, FPN, LinkNet, PAN, and X-Unet for semantic segmentation of UAV-acquired railway imagery. Another contribution is demonstrating these models on NVIDIA Jetson AGX Orin, combining high accuracy with low computational demand, making the approach viable for real-world infrastructure inspection. Processing speed was sufficient for real-time monitoring. This evaluation helps bridge the gap between algorithmic development and practical deployment, supporting future research on scalable UAV-based monitoring systems [3, 11].

## 3. Data and methods

### 3.1. Datasets

For the purposes of the research described in this article, a proprietary dataset of annotated RGB images depicting railway infrastructure and surrounding vegetation was developed. The images were acquired using unmanned aerial vehicles (UAVs) as part of work conducted at the Łukasiewicz Research Network – Institute of Aviation. Images were captured using a Sony ILX-LR1 RGB camera mounted on a DJI Matrice M600 UAV at altitudes between 20 and 120 meters under varying lighting and weather conditions. Each image has a resolution of $5280 \times 3956$ pixels, ensuring high detail for infrastructure and vegetation analysis.

The dataset consists of 1885 high-resolution annotated RGB images. The annotation process was carried out manually using the Label Studio environment, employing the `polygon` method, which enables precise object labeling. The annotations were performed by employees of the Łukasiewicz Research Network – Institute of Aviation, including the author of this paper.

Six semantic classes were defined for the annotation process, selected based on their relevance to the analysis of the railway infrastructure environment:

- *railway* – railway tracks and infrastructure elements,
- *trees*,

- *otherplants* – non-arboreal residual vegetation,
- *levelcrossing*,
- *background* – remaining areas irrelevant to the classification process.

The dataset was split into training and validation subsets in a 70 : 30 ratio, maintaining a representative class distribution across both partitions. For final evaluation, the validation subset was reused as the test set. This ensured models never saw test images during training. However, reusing the validation set as the test set does not preserve evaluation integrity, as a dedicated test set is needed for unbiased performance estimation.

Due to confidentiality restrictions imposed by the employer and project agreement, the dataset cannot be made publicly available. Representative sample images included in the training dataset are presented in Fig. 1.

### 3.2. Methods

The primary research method adopted in this study involved a comparative analysis of the accuracy of selected deep learning architectures in the task of semantic segmentation of RGB images. These images depict elements of linear infrastructure and the surrounding vegetation. The main focus was placed on railway infrastructure, including level crossings and the adjacent vegetation. This task is of particular importance in the context of automating technical inspections. It also concerns detecting areas where vegetation, especially taller forms such as trees, encroaches upon the *railway* right-of-way. Ultimately, such automatic detection is intended to support decision-making processes by railway infrastructure managers. Specifically, it helps in determining the necessity of clearing selected sections of railway lines from excessive overgrowth of tall vegetation.

As part of the conducted study, five deep neural network architectures representing different methodological approaches were compared. The selected models included: X-Unet, DeepLabV3+, Feature Pyramid Network (FPN), LinkNet, and Pyramid Attention Network (PAN). These architectures were chosen to ensure that the study encompassed both models focused on high-precision structural detail reconstruction, such as DeepLabV3+ and PAN, as well as lightweight solutions optimized for real-time performance, such as LinkNet.

All models were trained and evaluated within a unified experimental environment, which ensured consistency in data preprocessing, augmentation strategies, optimization methods, and performance reporting procedures.

Implementation used PyTorch Lightning with the segmentation_models.pytorch library. X-Unet was custom-built based on U-Net++ and nested U-Net concepts, adapted for multi-class semantic segmentation. All trained models were evaluated using standard segmentation quality metrics, including Pixel Accuracy, Precision, Recall, F1-score, and

Fig. 1. Sample images originating from the training dataset.

Intersection over Union (IoU), calculated separately for each class as well as reported as a weighted average across all classes.

All experiments were conducted under uniform conditions to ensure reliable comparison of architectures for UAV imagery analysis. Operational performance was also assessed during deployment on NVIDIA Jetson AGX Orin, including inference time, memory usage, and compliance with platform requirements.

### 3.2.1. Architectures of the models

This chapter provides an overview of the deep learning architectures investigated in this study, namely: DeepLabV3+, Feature Pyramid Network (FPN), X-Unet, LinkNet, and Pyramid Attention Network (PAN).

**DeepLabV3+**

The DeepLabV3+ architecture was designed with the goal of achieving high-precision semantic image segmentation. It represents an extension of earlier versions of the DeepLab model developed by Google Research, starting from DeepLabV1, through DeepLabV2 and DeepLabV3, up to DeepLabV3+. This architecture was the first to combine atrous convolutions with a complete decoder mechanism within an encoder-decoder structure [5, 6, 7].

A key feature shared across all versions of DeepLab is the use of atrous convolutions, which enable the expansion of the receptive field of filters without reducing the resolution of the feature maps or increasing the number of parameters. These modifications allow the model to capture broader spatial context, which is crucial for objects with variable scale, especially in UAV imagery captured at different altitudes. In the context of the present study, DeepLabV3+ proves well-suited for the analysis of elements such as trees, railway tracks, and trackside infrastructure.

In DeepLabV3, the Atrous Spatial Pyramid Pooling (ASPP) module was introduced as several parallel convolutional paths with different dilation rates, along with a global average pooling component. ASPP enables simultaneous analysis of local and global features, offering a rich contextual representation [6].

DeepLabV3+ adds a decoder that refines object boundaries by combining deep and shallow features. It uses upsampling, skip connections, and convolutional layers to integrate multi-level information, improving contour delineation, critical for segmenting tracks, crossings, and vegetation [7].

The DeepLabV3+ architecture can be configured with various encoder backbones. In the present study, the configuration employed a ResNet-50 encoder pretrained on the ImageNet dataset.

**Feature Pyramid Network (FPN)**

The next architecture discussed is the Feature Pyramid Network (FPN). Similar to DeepLabV3+, it is a convolutional architecture designed to effectively represent objects at multiple scales. FPN extends traditional CNNs, which often struggle with detecting objects of varying sizes. Initially introduced for object detection, it is now widely used in semantic segmentation [12, 15]. A distinctive feature of the Feature Pyramid Network (FPN) is its use of a bottom-up pathway and a top-down pathway, interconnected through lateral connections. The bottom-up pathway serves as a deep feature extractor, typically a backbone such as ResNet, which generates successive feature maps at

progressively lower spatial resolutions. The top-down pathway, in turn, performs progressive upsampling of high-level semantic feature maps, which are then merged with lower-level features via lateral connections. Lateral connections use $1 \times 1$ convolutions to align channel numbers, enabling efficient feature map summation. In semantic segmentation tasks, FPN is commonly used as a decoder-supporting component that facilitates the propagation of multi-scale feature maps, thereby improving the segmentation accuracy of small objects [16, 20]. This mechanism is crucial for segmenting vegetation, tracks, and crossings, especially when these elements vary in scale and position in UAV imagery. In this study, the FPN architecture was used with a ResNet-50 encoder pretrained on the ImageNet dataset. The model was adapted for the task of multi-class segmentation of RGB images captured by UAVs. The implementation leveraged the segmentation_models.pytorch library, which enabled standardization of the training process and facilitated direct comparison of results across different architectures. FPN is a lightweight architecture that can be deployed on low-power devices such as the NVIDIA Jetson AGX Orin [29]. This makes it a practical balance between computational cost and segmentation quality.

**LinkNet**

LinkNet is a lightweight encoder–decoder segmentation architecture designed with real-time performance in mind. It was first introduced by Chaurasia and Culurciello [4] as a solution optimized for both speed and segmentation accuracy. Due to its compact design and efficient use of residual connections, LinkNet has been successfully applied in various domains, including autonomous systems, edge devices, and UAV-based applications.

The core structure of LinkNet follows a classical encoder–decoder scheme, with a distinctive feature being the use of shortcut connections between corresponding encoder and decoder blocks. These connections transfer feature maps before downsampling, helping preserve spatial information and reduce loss during network propagation.

One of the key features of the LinkNet architecture is its encoder, which is based on a lightweight ResNet-class model. Each encoder block performs downsampling and feature extraction, with the resulting features passed to the corresponding decoder blocks. In the decoder, each block adds features from its encoder counterpart, applies $3 \times 3$ convolutions, and upsamples. By limiting operations and focusing on coarse features, LinkNet achieves high throughput with minimal accuracy loss. The architecture is capable of real-time performance even on edge platforms such as the NVIDIA Jetson TX2 and Jetson AGX Orin.

In this study, the LinkNet architecture was used with a ResNet-50 encoder pretrained on the ImageNet dataset. The network was implemented using the Python library segmentation_models.pytorch [9] to ensure consistent training and evaluation. LinkNet is particularly well-suited for deployment in edge computing systems, such as real-time UAV platforms. It serves as an onboard processing component where inference speed is as important as segmentation accuracy.

**Pyramid Attention Network (PAN)**

The Pyramid Attention Network (PAN) is a deep learning architecture designed for semantic image segmentation, with a particular focus on accurately capturing object structures and boundaries. The model was proposed by Li et al. in 2018 [14] as an extension of classical approaches such as PSPNet and DeepLab, which, despite their high effectiveness, do not sufficiently account for the spatial selectivity of features.

PAN builds on ResNet-based convolutional networks and extends the decoder for dense prediction tasks [14]. It includes two modules: Feature Pyramid Attention (FPA) and Global Attention Upsample (GAU). FPA processes encoder outputs using parallel convolutions with different kernel sizes and global average pooling to create an attention-enhanced spatial-semantic representation. GAU merges high-level features from FPA with detailed lower-level features, enabling effective context propagation in the decoder.

In this study, PAN was adapted for multi-class segmentation of RGB images acquired by UAVs. The variant used a ResNet-50 encoder pretrained on ImageNet and targeted classes such as *railway*, *trees*, *otherplants*, *levelcrossing*, and *background*. Implementation relied on segmentation_models.pytorch for seamless integration into the standardized training environment.

**X-Unet**

The X-Unet architecture is an extension of the U-Net architecture, incorporating a nested skip connections mechanism. It draws upon concepts introduced in UNet++ [31] and U-Net [21]. In this study, we employed an open-source implementation available in the lucidrains/x-unet repository, licensed under the MIT License [25].

The U-Net architecture was originally designed to be used in biomedical image segmentation [22]. Extensions like UNet++ [31] introduced dense hierarchical connections between encoder and decoder, improving segmentation performance. X-Unet builds on this by using a deep decoder and feature consolidation to integrate multi-level information.

The model includes a downsampling encoder and an extended upsampling decoder with nested, multi-level structures. Key mechanisms are nested skip connections, feature map consolidation, and modular design. Skip connections propagate features from deeper to higher layers, while consolidation aggregates multiple decoder paths for richer feature maps. The modular design allows adjusting nesting depth and channel count.

Unlike standard U-Net, X-Unet enables hierarchical, multi-stage feature propagation through its nested structure, improving fine detail representation and class separation along boundaries.

In this study, X-Unet was adapted for the task of multiclass segmentation on RGB images acquired from BSP. The version used a ResNet-50 encoder pretrained on ImageNet and was trained in PyTorch under the unified experimental protocol described in Subsection 3.2.

Tab. 1. Hyperparameters and Configuration of the Segmentation Models Used in the Study.

| Model | Backbone | LR | Weight Decay | Batch Size | Epochs | Early Stopping | Loss Functions | Scheduler |
|---|---|---|---|---|---|---|---|---|
| **DeepLabV3+** | ResNet-50 | 0.0001 | 1e-4 | 4 | 100 | patience = 10 | CrossEntropyLoss DiceLoss | ReduceLROnPlateau |
| **FPN** | ResNet-50 | 0.0001 | 1e-4 | 4 | 100 | patience = 10 | CrossEntropyLoss DiceLoss | ReduceLROnPlateau |
| **PAN** | ResNet-50 | 0.0001 | 1e-4 | 4 | 100 | patience = 10 | CrossEntropyLoss DiceLoss | ReduceLROnPlateau |
| **X-Unet** | Custom Encoder | 0.0001 | 1e-4 | 4 | 100 | patience = 10 | CrossEntropyLoss DiceLoss | ReduceLROnPlateau |
| **LinkNet** | ResNet-50 | 0.0001 | 1e-4 | 4 | 100 | patience = 10 | CrossEntropyLoss DiceLoss | ReduceLROnPlateau |

### 3.2.2. Implementation, training and evaluation procedures

All architectures utilized in this study were implemented using the PyTorch library. Each model followed a unified encoder-decoder scheme, with encoders based either on pre-trained ResNet-50 models (as in DeepLabV3+, FPN, PAN, and LinkNet) or on a dedicated convolutional encoder, as in the case of X-Unet. The decoder structure varied depending on the architecture, while the segmentation head was configured to perform classification across five analyzed object classes. For architectures using ResNet-50 encoders, weights were initialized from ImageNet pretraining, while the decoder and segmentation head were initialized randomly. The X-Unet model was trained from scratch with random weight initialization.

Training was performed on a Windows 10 desktop with an NVIDIA GeForce RTX 3080 GPU in a CUDA 11.6-enabled virtual environment. The Adam optimizer was used with an initial learning rate of 0.0001 and weight decay of 0.0001. A ReduceLROnPlateau scheduler halved the learning rate after five epochs without improvement in mean IoU, and early stopping was applied with a patience of 10 epochs.

All evaluated architectures were trained using the same pipeline. RGB images were rescaled and cropped to a resolution of $512 \times 512$ pixels and normalized to the $[0, 1]$ range. Data augmentation with Albumentations included 90° rotations, flips, contrast and brightness adjustments, and random cropping. Ground truth masks were encoded as integer label maps corresponding to six classes: *railway*, *trees*, *otherplants*, *levelcrossing*, and *background*.

Each model was trained for up to 100 epochs with a batch size of 2. The loss function combined DiceLoss and CrossEntropyLoss with equal weights (0.5 : 0.5). After each epoch, metrics such as pixel accuracy, IoU, mean IoU, precision, recall, and F1-score were computed. Models were saved based on the best mean IoU (mIoU) on the validation set.

All experiments used identical data splits, loss functions, metrics, and augmentation methods, ensuring an objective comparison of architectures. The hyperparameters and configuration data for segmentation models used are collected in Tab. 1.

Tab. 2. Evaluation Results for Pixel Accuracy, IoU, F1-score, Precision, and Recall for the DeepLabV3+ Architecture.

| Class | Pixel Accuracy | IoU | F1-score | Precision | Recall |
|---|---|---|---|---|---|
| 0 | 0.894730 | 0.796607 | 0.886791 | 0.878990 | 0.894730 |
| 4 | 0.942189 | 0.733874 | 0.846514 | 0.768478 | 0.942189 |
| 5 | 0.780494 | 0.632202 | 0.774662 | 0.768916 | 0.780494 |
| 9 | 0.931077 | 0.878396 | 0.935262 | 0.939485 | 0.931077 |
| 13 | 0.806878 | 0.702645 | 0.825357 | 0.844703 | 0.806878 |
| Mean | 0.867882 | 0.748745 | 0.853717 | 0.840115 | 0.871074 |

Tab. 3. Evaluation Results for Pixel Accuracy, IoU, F1-score, Precision, and Recall for the FPN Architecture.

| Class | Pixel Accuracy | IoU | F1-score | Precision | Recall |
|---|---|---|---|---|---|
| 0 | 0.909824 | 0.828377 | 0.906134 | 0.902473 | 0.909824 |
| 4 | 0.849934 | 0.795032 | 0.885814 | 0.924857 | 0.849934 |
| 5 | 0.793869 | 0.671752 | 0.803650 | 0.813676 | 0.793869 |
| 9 | 0.928339 | 0.886766 | 0.939985 | 0.951927 | 0.928339 |
| 13 | 0.878209 | 0.773510 | 0.872293 | 0.866456 | 0.878209 |
| Mean | 0.892011 | 0.791087 | 0.881575 | 0.891878 | 0.872035 |

## 4. Experimental results

This chapter presents the evaluation metric results for each of the architectures analyzed in the study. Summary tables Tab. 2 through Tab. 6 report the following metrics: Pixel Accuracy, IoU, F1-score, Precision, and Recall for the individual classes: *railway*, *levelcrossing*, *trees*, *otherplants*, and *background*. Additionally, average values of these metrics, aggregated across all classes, are provided. The tables correspond to the following architectures: DeepLabV3+ (Tab. 2), FPN (Tab. 3), LinkNet (Tab. 4), PAN (Tab. 5), and XUnet (Tab. 6).

Tab. 4. Evaluation Results for Pixel Accuracy, IoU, F1-score, Precision, and Recall for the LinkNet Architecture.

| Class | Pixel Accuracy | IoU | F1-score | Precision | Recall |
|---|---|---|---|---|---|
| 0 | 0.907080 | 0.785819 | 0.880066 | 0.854614 | 0.907080 |
| 4 | 0.949262 | 0.728692 | 0.843056 | 0.758224 | 0.949262 |
| 5 | 0.701903 | 0.596667 | 0.747391 | 0.799182 | 0.701903 |
| 9 | 0.951184 | 0.855406 | 0.922069 | 0.894683 | 0.951184 |
| 13 | 0.765662 | 0.688954 | 0.815835 | 0.873045 | 0.765662 |
| Mean | 0.858954 | 0.731107 | 0.841683 | 0.835949 | 0.855018 |

Tab. 5. Evaluation Results for Pixel Accuracy, IoU, F1-score, Precision, and Recall for the PAN Architecture.

| Class | Pixel Accuracy | IoU | F1-score | Precision | Recall |
|---|---|---|---|---|---|
| 0 | 0.734340 | 0.604364 | 0.753400 | 0.773476 | 0.734340 |
| 4 | 0.056374 | 0.052355 | 0.099500 | 0.423390 | 0.056374 |
| 5 | 0.316605 | 0.261133 | 0.414125 | 0.598463 | 0.316605 |
| 9 | 0.925309 | 0.525510 | 0.688963 | 0.548789 | 0.925309 |
| 13 | 0.610132 | 0.481592 | 0.650100 | 0.695673 | 0.610132 |
| Mean | 0.686672 | 0.384991 | 0.521218 | 0.607958 | 0.528552 |

Tab. 6. Evaluation Results for Pixel Accuracy, IoU, F1-score, Precision, and Recall for the XUnet Architecture.

| Class | Pixel Accuracy | IoU | F1-score | Precision | Recall |
|---|---|---|---|---|---|
| 0 | 0.886742 | 0.810324 | 0.895225 | 0.903872 | 0.886742 |
| 4 | 0.851897 | 0.793492 | 0.884857 | 0.920471 | 0.851897 |
| 5 | 0.749572 | 0.631012 | 0.773767 | 0.799578 | 0.749572 |
| 9 | 0.927654 | 0.876006 | 0.933905 | 0.940242 | 0.927654 |
| 13 | 0.889751 | 0.743043 | 0.852581 | 0.818392 | 0.889751 |
| Mean | 0.877694 | 0.770775 | 0.868067 | 0.876511 | 0.861123 |

For detailed analysis, it is advisable to focus on the following metrics: mean IoU, IoU per class, mean F1-score, and F1-score per class.The dataset is imbalanced, with the *levelcrossing* class notably underrepresented. These metrics allow for a more accurate interpretation of results.

The IoU per class metric measures how much of the actual area of a given class is correctly predicted by the model. This metric is crucial for assessing segmentation quality per class, including rare ones like *levelcrossing*. Without it, a high global score could mask poor detection of infrequent classes.

The mean IoU metric, on the other hand, represents the average IoU across all classes. It evaluates segmentation quality uniformly across classes, avoiding bias toward more frequent ones.

F1-score is the harmonic mean of precision and recall for each class. It captures both under- and over-segmentation, making it effective for detecting errors and handling classes with fewer pixels.

Mean F1-score is the average of F1-scores across all classes. It reflects the overall classification effectiveness regardless of class frequency and complements the mean IoU metric.

Table 7 is a summary of IoU metric results for all analyzed classes and for each of the deep learning architectures evaluated in this study. Additionally, the results for the mIoU

Tab. 7. Summary of IoU and Mean IoU Metric Results for Individual Models.

| IoU | DeepLabV3+ | FPN | LinkNet | PAN | Xunet |
|---|---|---|---|---|---|
| Class 0 | 0.796607 | 0.828377 | 0.785819 | 0.604364 | 0.810324 |
| Class 4 | 0.733874 | 0.795032 | 0.728692 | 0.052355 | 0.793492 |
| Class 5 | 0.632202 | 0.671752 | 0.596667 | 0.261133 | 0.631012 |
| Class 9 | 0.878396 | 0.886766 | 0.855406 | 0.525510 | 0.876006 |
| Class 13 | 0.702645 | 0.773510 | 0.688954 | 0.481592 | 0.743043 |
| Mean IoU | 0.748745 | 0.791087 | 0.731107 | 0.384991 | 0.770775 |

Tab. 8. Summary of F1-score and Mean F1-score Metric Results for Individual Models.

| F1-score | DeepLabV3+ | FPN | LinkNet | PAN | Xunet |
|---|---|---|---|---|---|
| Class 0 | 0.886791 | 0.906134 | 0.880066 | 0.753400 | 0.895225 |
| Class 4 | 0.846514 | 0.885814 | 0.843056 | 0.099500 | 0.884857 |
| Class 5 | 0.774662 | 0.803650 | 0.747391 | 0.414125 | 0.773767 |
| Class 9 | 0.935262 | 0.939985 | 0.922069 | 0.688963 | 0.933905 |
| Class 13 | 0.825357 | 0.872293 | 0.815835 | 0.650100 | 0.852581 |
| Mean F1-score | 0.853717 | 0.881575 | 0.84168 | 0.521218 | 0.868067 |

metric are also presented. As shown, the best performance in both per-class IoU and mIoU was achieved by the Feature Pyramid Network (FPN) architecture, followed by XUnet, DeepLabV3+, LinkNet, and finally Pyramid Attention Network (PAN), which obtained the lowest values for these metrics.

Table 8 presents the F1-score results for all classes and for each of the deep learning architectures evaluated in this study. Additionally, the results for the mean F1-score metric are also included. As can be observed, the best performance in both per-class F1-score and mean F1-score was achieved by the Feature Pyramid Network (FPN) architecture, followed by XUnet, DeepLabV3+, LinkNet, and finally Pyramid Attention Network (PAN), which obtained the lowest values for these metrics.

Figure 2 presents a graphical comparison of the mean IoU and mean F1-score values for the deep learning architectures analyzed in this study.

### 4.1. Model deployment on NVIDIA Jetson AGX Orin

The trained models were subsequently deployed on an NVIDIA Jetson AGX Orin unit to evaluate their performance in a real-time mission scenario. The deployment was made possible using the PyTorch Lightning library along with other Python libraries, including json, time, pathlib, albumentations, cv2, numpy, shapely, torch, skimage, pyvips, and PIL.

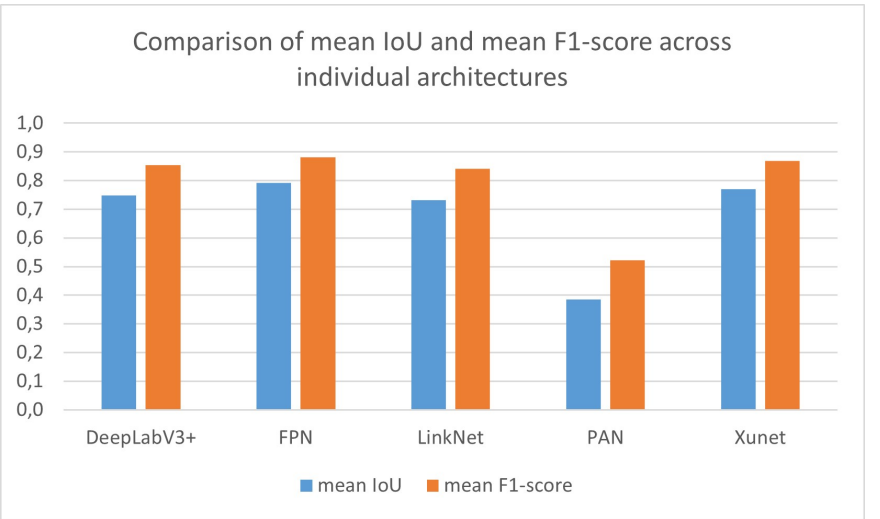


Fig. 2. Comparison of Mean IoU and Mean F1-score Values Across Individual Models.

The runtime code loads the trained model and is capable of analyzing incoming images in sequence. To achieve this, a monitoring mechanism was implemented to watch a designated folder where images from the ongoing mission are continuously saved. For each newly detected image, the system performs semantic segmentation using the classes defined during model training.

The results are generated in three formats: 1° as visualizations in which the predicted masks are overlaid semi-transparently on the original image, which also serves as a background for interpretation; 2° as JSON files saved in two separate directories, one for all analyzed images and another exclusively for alert cases, i.e., when the model detects dangerously close proximity between tall vegetation and railway tracks.

Each JSON file contains detailed information such as: sets of points outlining the contours of detected regions by class, class probability scores, the shortest distances between the *railway* and *trees* classes, flagged alert distances, the file path of the source image, timestamp, prediction time, image dimensions, an alert flag indicating whether a critical distance was detected, and the coordinates of the points that triggered the alert.

The visual and JSON outputs are then used for real-time visualization in a dedicated application designed specifically for this purpose.

All tested architectures were designed and implemented to ensure compatibility with deployment on the NVIDIA Jetson AGX Orin platform. Although inference was executed under conditions characteristic of real-time operation, no quantitative performance metrics (e.g., FPS, power consumption, energy efficiency) were analyzed. Therefore, the

contribution is limited to demonstrating deployment feasibility rather than providing a full performance evaluation. These aspects represent an important direction for future research focused on optimizing embedded inference.

### 4.2. Model performance results in a real-world field mission

Figure 3 presents a sample of input data in the form of RGB images acquired from a camera mounted on a UAV unit, along with a sample of output data in the form of visual files showing the results produced by the trained model. These visualizations illustrate how the model assigned detected regions to the appropriate classes. According to the adopted color scheme, the *railway* class is marked in green, *otherplants* in purple, and *trees* in pink.

## 5. Conclusions

The conducted research successfully achieved the key scientific objectives. Five models based on deep learning architectures were trained: DeepLabV3+, FPN, X-Unet, LinkNet, and PAN. The training conditions and parameters were carefully selected to enable a reliable comparison of the individual architectures. The comparison used metrics such as Pixel Accuracy, IoU, F1-score, Precision, and Recall, computed per class and averaged across all classes, with emphasis on mean IoU and mean F1-score as key indicators. FPN achieved the highest performance, followed by X-Unet, DeepLabV3+, LinkNet, and PAN.

The trained models were successfully deployed on the target platform, an NVIDIA Jetson AGX Orin module mounted onboard the DJI Matrice M600 UAV. Deployment required appropriate adaptation of the runtime environment.

The best-performing models accurately captured features in RGB images, correctly delineating object classes with precise contours. Real-time visualizations were generated directly on the Jetson AGX Orin onboard the UAV.

In addition to these findings, the study highlights a trade-off between segmentation accuracy and computational efficiency. FPN achieved the highest accuracy, while X-Unet delivered competitive results with lower complexity, making it suitable for real-time embedded deployment. These insights suggest that architecture selection should reflect application priorities: high accuracy for offline analysis versus lightweight models for UAV-based real-time monitoring. The proposed approach also shows potential for adaptation to other types of linear infrastructure, such as roads or power lines, with appropriate adjustments.

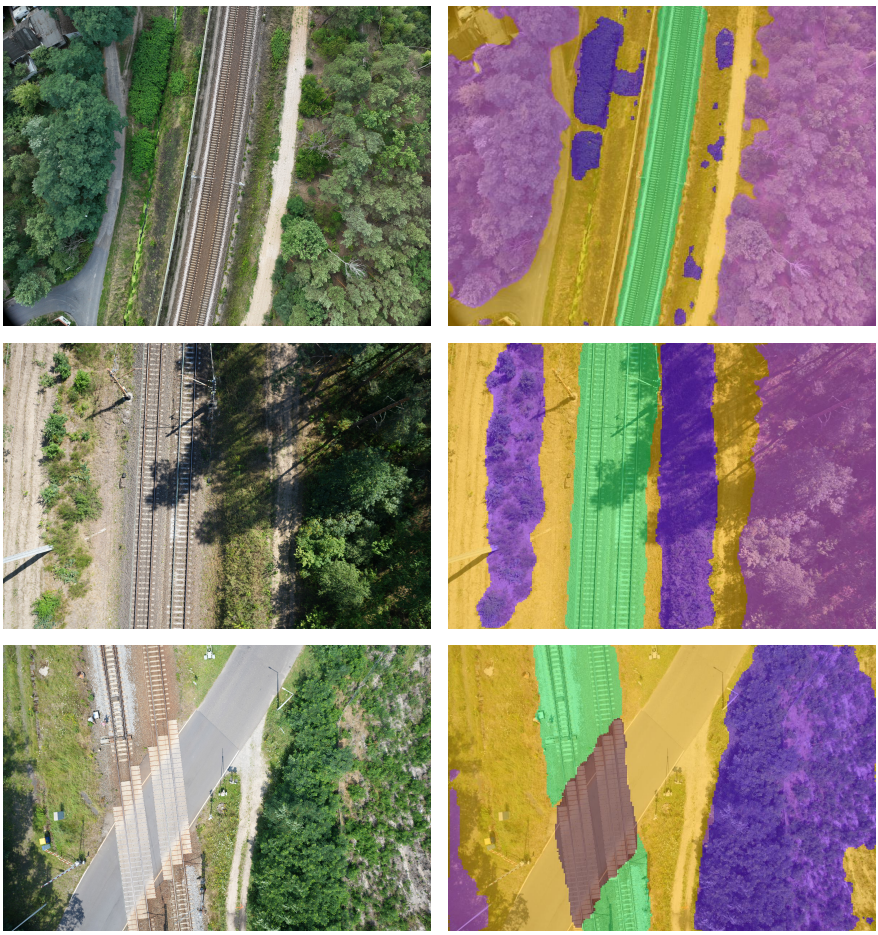

Fig. 3. Input images (on the left) and output results presented as visualizations (on the right). The results pertain to the XUnet architecture.

## 6. Discussion of limitations

The experimental results revealed differences in performance among the tested architectures. FPN achieved the highest mean IoU (0.791) and mean F1-score (0.882), indicating superior segmentation quality across all classes. This can be attributed to its multi-scale feature aggregation strategy, which effectively captures both global and local context. DeepLabV3+ also delivered strong results (mIoU = 0.749, mean F1-score = 0.854), benefiting from the Atrous Spatial Pyramid Pooling module, though its computational

complexity makes it less suitable for resource-constrained environments. X-Unet demonstrated competitive accuracy (mIoU = 0.771, mean F1-score = 0.868) while maintaining a lightweight structure, which is particularly advantageous for real-time deployment on embedded platforms such as NVIDIA Jetson AGX Orin. LinkNet provided moderate performance (mIoU = 0.731, mean F1-score = 0.842), representing a compromise between accuracy and efficiency. In contrast, PAN obtained the lowest scores (mIoU = 0.385, mean F1-score = 0.521), primarily due to difficulties in segmenting rare classes like level crossings, suggesting that attention-based mechanisms alone may not suffice without robust multi-scale processing. These findings highlight a trade-off between segmentation accuracy and computational efficiency, underlining the importance of selecting architectures based on application-specific priorities: high accuracy for offline analysis versus lightweight models for real-time UAV monitoring.

Several aspects can be identified as limitations of the conducted study. First, the set of analyzed architectures could be expanded, or further research could be carried out to improve the performance of the currently evaluated models.

Second, the size of the training, validation, and test datasets could be increased. Railway infrastructure, particularly the surrounding vegetation, is highly complex, with a vast diversity of plant cover types and railway embankment structures. Expanding these datasets and annotating additional images would likely enhance the models' generalization capabilities, thereby improving the overall quality of the developed solution.

Third, a more powerful computational unit than the NVIDIA Jetson AGX Orin could be employed. Developing and utilizing a more efficient processing platform could lead to faster input image processing and result generation.

Fourth, the proposed approach could be transferred to other domains and applications. The models can be trained to monitor other types of linear infrastructure, such as roads, power lines, rivers, or shorelines. With the current solution already developed, it is feasible to adapt it to new use cases, provided that appropriate adjustments are made.

Fifth, reusing the validation set as the test set does not preserve the integrity of the evaluation, as a dedicated test set is required for an unbiased performance estimate. This methodological limitation should be considered when interpreting the reported results.

Sixth, the study did not include a quantitative analysis of performance metrics (e.g., FPS, latency, power consumption) for embedded deployment. Therefore, the contribution is limited to demonstrating deployment feasibility rather than providing a full performance evaluation or confirming real-time operation. This aspect may represent an important direction to consider in future research aimed at strengthening the practical relevance of the proposed approach.

Under the current requirements, the developed solution fully met the project objectives and paved the way for broader research in this area. The conducted study constitutes a solid foundation for further work, both within the current scope and in related fields where automated monitoring of linear infrastructure is required.

**Acknowledgement**

This research was supported by the Łukasiewicz Research Network – Institute of Aviation in the project entitled *Assessment and analysis of the potential use of remote object detection methods and condition identification of critical infrastructure* carried out by the Remote Sensing Department. The author expresses her gratitude for the support provided.

## References

[1] P. Aela, H.-L. Chi, A. Fares, T. Zayed, and M. Kim. UAV-based studies in railway infrastructure monitoring. *Automation in Construction* 167:105714, 2024. doi:10.1016/j.autcon.2024.105714.

[2] P. Anilkumar, P. Venugopal, K. Lokesh, G. NagaJyothi, and M. Nanda kumar. AA-TransDeepLabv3+: a novel semantic segmentation framework for aerial images using adaptive and attentive based Transdeeplabv3+ with hybrid optimization technique. *Signal, Image and Video Processing* 19(225), 2025. doi:10.1007/s11760-024-03617-z.

[3] A. Chandramouli, H. Song, M. Liu, a. Damai, H. S. Narman, et al. Deep learning approaches for railroad infrastructure monitoring: Comparing YOLO and vision transformers for defect detection. In: *2025 IEEE 16th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, pp. 0205–0211, 2025. doi:10.1109/UEMCON67449.2025.11267623.

[4] A. Chaurasia and E. Culurciello. LinkNet: Exploiting encoder representations for efficient semantic segmentation. In: *2017 IEEE Visual Communications and Image Processing (VCIP)*, pp. 1–4, 2017. doi:10.1109/VCIP.2017.8305148.

[5] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40(4):834–848, 2018. doi:10.1109/TPAMI.2017.2699184.

[6] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. arXiv, arXiv:1706.05587, 2017. doi:10.48550/arXiv.1706.05587.

[7] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 833–851, 2018. doi:10.1007/978-3-030-01234-2_49.

[8] C. Chenglin, W. Fei, Y. Min, Q. Yong, and B. Yun. Edge-enabled real-time railway track segmentation. arXiv, arXiv:2401.11492, 2024. doi:10.48550/arXiv.2401.11492.

[9] S. Chilamkurthy. segmentation_models.pytorch: Segmentation models. Python library with neural networks for image segmentation based on PyTorch. GitHub, 2019. `https://github.com/chsasank/segmentation_models.pytorch`. [Accessed: 2024].

[10] M. Di Summa, M. E. Griseta, N. Mosca, C. Patruno, M. Nitti, et al. A review on deep learning techniques for railway infrastructure monitoring. *IEEE Access* 11:114638–114661, 2023. doi:10.1109/ACCESS.2023.3309814.

[11] M. Giunta, V. Barrile, G. Leonardi, and E. Genovese. Comprehensive railway track monitoring using unmanned aerial systems (UASs) and building information modelling (BIM). In: *Computational Science and Its Applications – ICCSA 2025 Workshops*, vol. 15894 of *Lecture Notes in Computer Science*, pp. 407–419. Springer, 2025. doi:10.1007/978-3-031-97648-3_27.

[12] A. Kirillov, R. Girshick, K. He, and P. Dollár. Panoptic feature pyramid networks. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6392–6401, 2019. doi:10.1109/CVPR.2019.00656.

[13] Y. Kwon, W. Kim, and H. Kim. HARD: Hardware-aware lightweight real-time semantic segmentation model deployable from edge to GPU. In: *Computer Vision – ACCV 2024*, Lecture Notes in Computer Science, pp. 252–269, 2024. doi:10.1007/978-981-96-0963-5_15.

[14] H. Li, P. Xiong, J. An, and L. Wang. Pyramid attention network for semantic segmentation. arXiv, arXiv:1805.10180, 2018. doi:10.48550/arXiv.1805.10180.

[15] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, et al. Feature pyramid networks for object detection. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 936–944, 2017. doi:10.1109/CVPR.2017.106.

[16] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia. Path aggregation network for instance segmentation. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8759–8768, 2018. doi:10.1109/CVPR.2018.00913.

[17] M. Lopez-Montiel, D. A. Lopez, and O. Montiel. JetSeg: Efficient real-time semantic segmentation model for low-power GPU-embedded systems. arXiv, arXiv:2305.11419, 2023. doi:10.48550/arXiv.2305.11419.

[18] Y.-H. Na and D.-K. Kim. Deep learning strategy for UAV-based multi-class damage detection on railway bridges using U-Net with different loss functions. *Applied Sciences* 15(15):8719, 2025. doi:10.3390/app15158719.

[19] C. R. Nagarathna. Intelligent aerial surveillance for safer railways using machine learning. *International Journal of Innovative Research and Scientific Studies* 8(5):1160–1166, 2025. doi:10.53894/ijirss.v8i5.9077.

[20] S. Qiao, L.-C. Chen, and A. Yuille. DetectoRS: Detecting objects with recursive feature pyramid and switchable atrous convolution. In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10208–10219, 2021. doi:10.1109/CVPR46437.2021.01008.

[21] X. Qin, Z. Zhang, C. Huang, M. Dehghan, O. R. Zaiane, et al. U2-Net: Going deeper with nested U-structure for salient object detection. *Pattern Recognition* 106:107404, 2020. doi:10.1016/j.patcog.2020.107404.

[22] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional networks for biomedical image segmentation. In: *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 234–241, 2015. doi:10.1007/978-3-319-24574-4_28.

[23] C. Shen, J. Zhang, Y. Ji, T. Xu, L. Jiang, et al. Real-time semantic segmentation for UAV perspectives on embedded platforms. In: *Advanced Intelligent Computing Technology and Applications. ICIC 2025*, vol. 15842 of *Lecture Notes in Computer Science*, pp. 425–434. Springer, 2025. doi:10.1007/978-981-96-9863-9_36.

[24] K. Stypułkowski, P. Gołda, K. Lewczuk, and J. Tomaszewska. Monitoring system for railway infrastructure elements based on thermal imaging analysis. *Sensors* 21(11):3819, 2021. doi:10.3390/s21113819.

[25] P. Wang. x-unet: Implementation of a U-net complete with efficient attention as well as the latest research findings. GitHub, 2024. `https://github.com/lucidrains/x-unet`. [Accessed: 2024].

[26] L. Wen, Y. Peng, M. Lin, N. Gan, and R. Tan. Multi-modal contrastive learning for LiDAR point cloud rail-obstacle detection in complex weather. *Electronics* 13(1):220, 2024. doi:10.3390/electronics13010220.

[27] Y. Weng, Z. Li, X. Chen, J. He, F. Liu, et al. A railway track extraction method based on improved DeepLabV3+. *Electronics* 12(16):3500, 2023. doi:10.3390/electronics12163500.

[28] Y. Weng, J. Yang, C. Zhang, J. He, C. Peng, et al. An improved DeepLabv3+ railway track extraction algorithm based on densely connected and attention mechanisms. *Scientific Reports* 15:2556, 2025. doi:10.1038/s41598-024-84937-5.

[29] M. Xu, Y. Guo, and J. Luo. Lightweight feature pyramid networks for real-time semantic segmentation on edge devices. *IEEE Access* 10:33645–33655, 2022. doi:10.1109/ACCESS.2022.3161230.

[30] Z. Zhang and G. Li. UAV imagery real-time semantic segmentation with global–local information attention. *Sensors* 25(6):1786, 2025. doi:10.3390/s25061786.

[31] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang. UNet++: A nested U-Net architecture for medical image segmentation. In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support (DLMIA 2018)*, pp. 3–11, 2018. doi:10.1007/978-3-030-00889-5_1.

# Application of Computer Vision Technology in the Recognition of Guzheng Playing Posture

Dan Lu
*College of Art, Northeast Agricultural University, Harbin, Heilongjiang, China*
**Corresponding author: Dan Lu (ludan_vip@outlook.com)*

**Submitted:** 24 Jun 2025 **Accepted:** 03 Oct 2025 **Published:** 31 Mar 2026 

**Abstract** This study addresses the performance teaching needs of traditional Chinese Guzheng and attempts to introduce computer vision and deep learning technologies into gesture recognition tasks. By constructing a dataset that includes various Guzheng playing actions, image sequences are collected during the performance process. Combined with convolutional neural networks for feature extraction, this approach achieves automatic recognition of multiple basic gestures. The model employs an optimized ResNet50 structure, maintaining high recognition accuracy under standardized image input and weighted classifiers. Experiments show that the system performs stably in recognizing typical actions and has a certain tolerance for complex action transitions and partial hand occlusions. When deployed in educational settings, the system can provide real-time feedback and visual presentations, assisting teachers in evaluating students' gesture standards and enhancing interactive teaching effects. From the perspective of engineering implementation and practicality in education, this research provides methodological support for the integration of traditional arts and artificial intelligence, laying the groundwork for future intelligent musical instrument training systems. Overall results indicate that this technical approach holds practical significance and application potential in improving Guzheng performance quality and reducing teaching costs.

**Keywords:** computer vision technology, posture recognition, Guzheng playing, intelligent teaching system.

## 1. Introduction

The Guzheng, as a traditional Chinese ethnic instrument, boasts a long history of development and a rich array of playing techniques. In actual performance, the posture not only affects sound production but also directly impacts the player's hand health and technical stability. Traditional Guzheng instruction primarily relies on teacher demonstrations and verbal guidance, which can be subjective and result in delayed feedback. With the rapid advancement of artificial intelligence technology, computer vision has provided new tools for posture recognition and action analysis. By capturing performance actions in real-time through visual systems, automatic recognition and evaluation of postures can be achieved, aiding in performance training. Computer vision has shown excellent application potential in fields such as sports and rehabilitation medicine, but it is still in its early stages of exploration in music education. Applying computer vision to the recognition of Guzheng playing postures not only helps enhance the level of intelligent teaching but also provides data support for scientific analysis of performance

actions. Through technical means, standardized posture modeling, detection of abnormal movements, and optimization of playing habits can be realized, offering significant theoretical research value and practical application significance.

In recent years, the application of computer vision technology in various fields has continuously expanded, providing crucial support for the intelligent transformation of traditional industries. Shan et al. (2025) [18] reviewed the application of computer vision in sustainable mining engineering, noting that image recognition and automated monitoring have improved mine operation efficiency and safety. Gill et al. (2024) [6] analyzed the integration methods of computer vision under Industry 4.0, arguing that it can effectively promote production process optimization and intelligent resource allocation. Kataev and Bulysheva (2024) [11] proposed using computer vision for automatic defect detection in ceramic tiles, which can achieve efficient identification of minor defects and enhance product quality control levels. Hui and Geng (2024) [10] explored the construction of an intelligent English mixed teaching system in a 5G environment, emphasizing the critical role of computer vision in real-time interaction and learning behavior recognition.

Huang et al. (2024) [8] reviewed the development of trustworthy computer vision from the perspective of ethics and technological evolution, highlighting that enhancing model transparency and reliability is a crucial direction for future research. Blose and Schenkel (2024) [2] found through facial and body posture emotion recognition studies that posture features play an independent and significant role in emotion decoding, emphasizing the necessity of fine-grained modeling in action recognition. Li and Chen (2023) [13] demonstrated in their study on robot English translation based on computer vision that visual technology can enhance the understanding and transmission of linguistic and cultural information. Yin (2023) [25] analyzed cross-cultural competence development in Guzheng education, pointing out that gesture recognition and feedback mechanisms supported by MOOC platforms can improve teaching effectiveness and professional competence.

Upadhyay et al. (2023) [23] proposed a deep learning-based yoga pose recognition model, validating the effectiveness of convolutional networks in complex pose classification and providing algorithmic support for music gesture recognition. Shih et al. (2023) [19] designed an intelligent math tutoring system based on diagnostic teaching, emphasizing the potential of combining computer vision with cognitive modeling in adaptive teaching. Huang et al. (2022) [9] analyzed learners' human pose recognition behavior under the context of maker education, proposing that there is a significant correlation between pose data and learning outcomes. Valipoor and de Antonio (2023) [24] systematically reviewed the development trends of scene understanding based on computer vision in the field of visual assistance for the blind, highlighting the importance of multimodal perception and human-computer interaction optimization. Existing research clearly demonstrates the broad application potential of computer vision technology in

action recognition, intelligent teaching, cultural interaction, and decision support [20]. However, specialized studies on Guzheng performance pose recognition are still insufficient, and the fine-grained analysis of model actions and real-time feedback systems need further development.

In traditional Guzheng teaching, the evaluation of performance posture mainly relies on teachers' experience, lacking objective and systematic quantitative standards. This approach struggles to ensure the stability and accuracy of teaching quality when dealing with large numbers of students or beginners. Although some existing studies have introduced motion capture systems, they generally suffer from issues such as expensive equipment, complex operation, and interference with natural performances, making it difficult to promote their application in actual teaching. Research on Guzheng posture recognition based on computer vision is relatively scarce, lacking targeted models and systematic methods, which results in insufficient accuracy and applicability. This study aims to develop an efficient and low-intrusive method for Guzheng performance posture recognition using deep learning and image processing technologies. It involves constructing a standard dataset, designing a visual recognition model that adapts to musical action features, and achieving automatic recognition and evaluation of performance posture. The goal is to break through existing technical bottlenecks through this research, providing an intelligent auxiliary system for Guzheng teaching and performance training, promoting the deep integration of traditional arts and modern technology.

The study employs a computer vision-based pose recognition method, combined with a deep learning framework for model construction and training. First, high-definition cameras from multiple angles capture pose images during the performance process to establish a dataset that includes various playing postures. Then, a convolutional neural network (CNN) is used as the foundation to design a lightweight and adaptable visual recognition model. To improve recognition accuracy, feature enhancement modules and attention mechanisms are introduced to optimize the feature extraction process. During the model training phase, data augmentation and transfer learning techniques are applied to enhance the model's generalization ability. The overall system workflow includes data collection, image preprocessing, feature extraction, pose classification, and feedback output. Finally, the system performance and application effects are validated through actual performance tests. Throughout the research, emphasis is placed on the practicality and scalability of the methods, ensuring that the proposed pose recognition approach can adapt to different playing styles and individual differences, providing technical support for the scientific analysis and intelligent evaluation of Guzheng playing postures.

## 2. Materials and methods

### 2.1. Data collection and sample construction

#### 2.1.1. Selection criteria and sample basic information of performers

To ensure the diversity and representativeness of the training data for the posture recognition model, performers are selected based on certain criteria. First, performers must have over one year of Guzheng performance experience and be proficient in basic finger techniques and common repertoire playing skills. Second, participants must not have significant upper limb movement disorders to ensure that their movements are natural, smooth, and can be standardized for recording. During the sample selection process, gender, age, and performance level diversity is considered to cover different body postures and styles of movement. Ultimately, 20 performers were selected, with a roughly balanced gender ratio and ages ranging from 18 to 35 years old, and performance experience spanning from 1 to 10 years. All participants received standardized movement guidance before data collection to minimize individual differences that could cause errors, ensuring the usability and consistency of the collected data. Before and after data collection, identity information was anonymized and data was encrypted to protect the privacy of the participants. The specific basic information of the samples is shown in Tab. 1.

To capture stylistic variability, we stratified recruitment by three common performance styles: floor-level style (instrument low and performer seated on the floor), seated classical style (performer seated in front of a standard stand-mounted Guzheng), and standing style (elevated instrument with the performer standing). Among 20 participants, the distribution was 6 floor-level, 10 seated classical, and 4 standing. During collection, seat/stand height and instrument angle were recorded to contextualize posture geometry. Style labels are included in the metadata and enable subgroup analyses. On the held-out test set, class-balanced accuracy by style was 92.4% (floor-level), 93.1%

Tab. 1. Statistics of basic information on performers.

| Number of performers | sex | age | Duration of playing (years) | Lead acting style |
|---|---|---|---|---|
| 1 | woman | 22 | 3 | Orthodox tradition |
| 2 | man | 28 | 5 | Genre fusion |
| ... | ... | ... | ... | ... |
| 20 | woman | 25 | 7 | Genre fusion |

(seated classical), and 91.0% (standing), suggesting minor viewpoint-related occlusions in standing performances but overall robust generalization.

### 2.1.2. Configuration of attitude image acquisition system

To obtain high-quality performance posture data, a professional image acquisition system was established. The system employs a dual-camera synchronous acquisition mode, recording the performer's movements from both front and side angles. Industrial-grade high-definition cameras with a resolution of $1920 \times 1080$ pixels and a sampling rate of 60 frames per second are used to ensure the precision and continuity of motion capture. The lens focal length is set at 35 mm to capture both local details and overall posture clearly. The acquisition environment is arranged under natural light conditions, with auxiliary lighting to ensure uniform brightness and low noise levels in the images. All equipment is connected via USB 3.0 interfaces to ensure efficient and stable data transmission. The synchronization control module coordinates the consistency of the two camera signals, preventing data discrepancies caused by time delays [15]. The acquisition system is equipped with a stable stand and standardized background cloth to minimize environmental interference affecting recognition accuracy. Hardware parameters of the acquisition system are listed in Tab. 2.

Both cameras were triggered synchronously to capture paired frontal and lateral views. For model training and evaluation reported here, we treat each frame as an independent sample and use the frontal view as the primary input to the single-image network. The lateral view served two purposes:

(i) data diversity — selecting lateral frames into the training pool at a 1 : 3 ratio to improve view robustness;
(ii) and cross-view validation — testing generalization when viewpoint shifts.

Tab. 2. Hardware equipment parameters.

| Device Name | Model | Resolution Ratio | Frame Rate | Interface Type |
|---|---|---|---|---|
| Main camera | Basler acA1920-155 um | 1920×1080 | 60 fps | USB3.0 |
| Auxiliary camera | Basler acA1920-155 um | 1920×1080 | 60 fps | USB3.0 |
| Isochronous controller | IDS Sync Controller | – | – | USB |
| Data acquisition server | Dell Precision 5820 | – | – | Gigabit network |

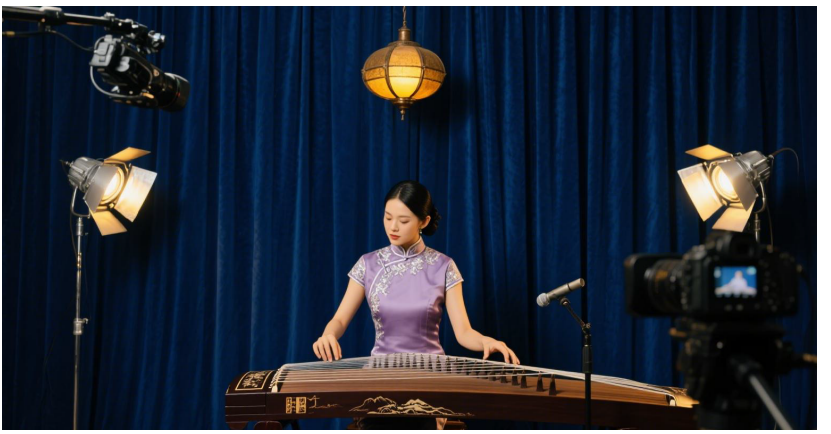

Fig. 1. The test site.

In optional deployments where both streams are available, we implement late fusion by averaging posterior probabilities from two identical single-frame models; the main results in this paper, however, are based on the frontal stream to ensure comparability across sessions.

To improve methodological transparency, we add a site photograph of the data-collection setup. The frontal camera was positioned 1.8 m from the performer at a height of 1.25 m, normal to the Guzheng's long edge; the lateral camera was 1.5 m from the right side at a height of 1.15 m with a yaw of about 80°. Two softbox lamps (500 lux to 700 lux at the soundboard) provided uniform illumination while avoiding specular glare. A matte dark backdrop minimized background clutter. Tripods were marked on the floor to keep geometry fixed across sessions. Fig. 1 shows the Guzheng centered on a marked area, the frontal and lateral cameras on tripods, softbox lighting, synchronization controller, and the performer in the standardized posture used for calibration.

### 2.1.3. Data preprocessing methods

To improve the efficiency and accuracy of pose feature extraction in the model, all collected image data undergoes standardized preprocessing before inputting into the model. The preprocessing process mainly includes image cropping, size normalization, brightness equalization, and data augmentation. Images are automatically cropped based on the playing area to remove irrelevant background interference. Subsequently, images are uniformly resized to $224 \times 224$ pixels to ensure consistent input dimensions. To mitigate the difficulty of feature extraction caused by lighting changes, adaptive histogram equalization is used to normalize the brightness of the images. Data augmentation introduces random rotation, horizontal flipping, and mild Gaussian noise perturbations to

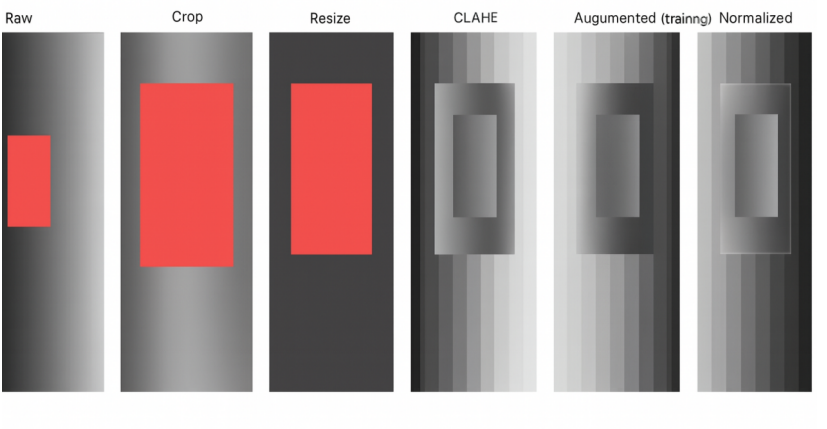


Fig. 2. Preprocessing pipeline, single frame.

increase sample diversity and enhance model robustness. Finally, all image pixel values are normalized to the $[0, 1]$ range to facilitate faster model convergence [22]. The specific normalization process is described as

$$X' = \frac{X - \min(X)}{\max(X) - \min(X)}\,, \tag{1}$$

where $X$ represents the original pixel matrix, and $\min(X)$ and $\max(X)$ respectively represent the minimum and maximum values of sample pixels in the testing set of images. Through standardization, the stability of the training stage and the final recognition accuracy are effectively improved.

To make preprocessing auditable, we visualize each step on the same sample frame. Starting from the raw frame, we apply:

(a) Region-of-interest crop anchored on the instrument's soundboard contour (margin 8–10% of width);
(b) Resize to $224\times224$ pixels with bilinear interpolation;
(c) Illumination normalization via CLAHE (`clipLimit` 2.0, `tileGridSize` $8\times8$);
(d) Color jitter for augmentation (brightness $\pm10\%$, contrast $\pm10\%$ during training only);
(e) Horizontal flip with $p = 0.5$ (training only; disabled for evaluation);
(f) Light Gaussian noise $\delta = 0.01$;
(g) Min–max scaling to $[0, 1]$.

As shown in Fig. 2, Panels show Raw → Crop → Resize → CLAHE → Augmented (training) → Normalized, enabling visual inspection of the cumulative effects before model ingestion.

Tab. 3. Comparison of performance of mainstream algorithms.

| Model name | Top-1 precision [%] | Number of parameters [M] | Speed of reasoning [ms/image] |
|---|---|---|---|
| VGG16 | 71.5 | 138 | 25 |
| ResNet50 | 76.2 | 25.6 | 15 |
| MobileNetV2 | 72 | 3.5 | 10 |
| EfficientNet-B0 | 77.1 | 5.3 | 12 |

## 2.2. Model construction and training strategy

### 2.2.1. Analysis of visual model selection basis

In the task of recognizing Guzheng performance postures, achieving both high recognition accuracy and inference efficiency is crucial. Model selection must be systematically considered from multiple dimensions. The accuracy of the model, parameter size, computational efficiency, and scalability collectively form the evaluation criteria. Currently, typical deep learning models widely used in image recognition tasks include the VGG series, ResNet series, MobileNet series, and the EfficientNet, which has shown outstanding performance in recent years. VGG16, as a representative of early deep convolutional networks, despite its clear structure and ease of implementation, has a large number of parameters and slow inference speed, making it unsuitable for lightweight requirements. ResNet50 introduces residual structures to effectively alleviate gradient disappearance issues in deep networks and has achieved excellent performance on various image recognition benchmark tests, demonstrating good accuracy and controllable complexity. MobileNetV2 employs depth separable convolutions, resulting in an extremely lightweight model suitable for deployment on mobile devices. Although its accuracy is slightly lower, it offers significant advantages in computational efficiency. EfficientNet achieves a better balance between model size, accuracy, and speed, making it particularly suitable for resource-constrained deployment environments such as [21].

In order to ensure the identification effect, the study needs to achieve near real-time feedback, so ResNet50 was selected as the benchmark model, and the subsequent lightweight and optimization strategies were combined to achieve the unity of high performance and high availability. Comparative data of four mainstream models is shown in Tab. 3.

### 2.2.2. Network architecture design details

Network architecture design is crucial for the successful recognition of posture actions in computer vision systems. This study has customized the ResNet50 infrastructure to better align with the visual characteristics of Guzheng playing movements. In the original

ResNet structure, residual connections serve as the main pathway, deepening network depth without increasing computational costs, thereby enhancing the model's stability and expressiveness in multi-layer semantic feature extraction. To improve the perception of local changes in the hand, the study adjusted the size of the first convolutional kernel from $7\times7$ to $5\times5$, making the network more sensitive to capturing local action details at the initial stage. In subsequent network layers, a strategy of stacking small-sized convolutional kernels was adopted, along with the addition of batch normalization layers, to accelerate model convergence and mitigate overfitting risks [17].

The network is optimized by introducing lightweight convolutional structures such as Depthwise Separable Convolution, which compress redundant computational paths and reduce resource consumption. At the network's end, a multi-layer perception fusion structure (multi-level feature fusion) is connected, combining low-level local features with high-level semantic features to enhance the modeling capability for complex pose variations. In forward propagation, the convolution operation serves as the basic unit for feature extraction, and its calculation method is described as

$$Y(i,j) = \sum_m \sum_n X(i+m, j+n) \times K(m,n), \tag{2}$$

where $X$ represents the input image, $K$ is the convolution kernel, $Y$ is the output feature map, $i, j$ are pixel indices, and $m, n$ are the translation offsets of the convolution window. By continuously optimizing the convolution kernel through iterative processes, the network can automatically learn stable gesture and pose features from large amounts of image data.

### 2.2.3. Feature extraction and pose classifier implementation

The effectiveness of the pose recognition model depends on the sufficiency of feature extraction and the classifier's ability to distinguish fine-grained actions. After the deep convolutional network constructed in the previous section extracts multi-scale spatial features, it needs to map high-dimensional feature vectors to a finite space of pose categories to complete the task of recognizing performance actions. For this purpose, this study designs a two-layer classifier module. The first layer is a 512-dimensional fully connected layer with ReLU as the activation function; the second layer is a Softmax output layer with the number of nodes equal to the number of pose categories, and the output represents the probability distribution for each category. The overall architecture parameters of the classifier are streamlined to enhance inference speed and facilitate integration into [5].

In the training process, cross entropy is used as the main loss function, and the problem of uneven sample distribution is considered. The category weight mechanism is introduced to balance and optimize the dominant categories and scarce categories in

the training process. The mathematical form of the loss function is

$$L = -\sum_{i=1}^{C} w_i y_i \log(\hat{y}_i), \tag{3}$$

where $C$ represents the total number of categories, $w_i$ is the weight of the sample in category $i$, $y_i$ is the actual label, and $\hat{y}_i$ is the probability value predicted by the model. The weight $w_i$ is designed based on the inverse of the frequency of each category, effectively mitigating the negative impact of skewed distribution of pose categories in the training set on learning performance. Through this strategy, the model not only maintains overall accuracy but also significantly enhances its ability to recognize a few complex poses.

#### 2.2.4. Integration of attitude recognition system

The ultimate goal of the Guzheng performance posture recognition system is to achieve a complete end-to-end recognition and feedback process. Therefore, the system integration phase is particularly critical after the model design is completed. The recognition system developed in this study consists of four main modules: the front-end data acquisition module, the intermediate image processing and model inference module, the back-end posture evaluation module, and the feedback visualization interface. The front end captures real-time image data through a camera and performs standardization processing. The middle part uses trained deep networks to complete feature extraction and posture classification. The back-end system converts the model prediction results into standard posture labels and generates corresponding prompt information, which is presented in real-time via a visual component on [12].

The system achieves efficient data transmission between modules through the Socket communication mechanism, complemented by edge computing devices for low-latency deployment. In practical teaching scenarios, the system can be embedded in smart piano desks or performance classroom terminals to analyze performers' movements in real-time and provide posture scores and suggestions, assisting in teaching and enhancing training efficiency. The complete integration process of the system is shown in Fig. 3.

### 2.3. Training and verification

#### 2.3.1. Model training parameter setting

The parameter settings during the model training process directly impact the final recognition performance and convergence efficiency. To achieve stable training and higher generalization capabilities, this study systematically optimized the hyperparameters of the network training. First, regarding the learning rate, a dynamic decay strategy with an initial value of 0.001 was adopted. The learning rate was dynamically adjusted based on the loss in the validation set to avoid oscillatory convergence or premature local

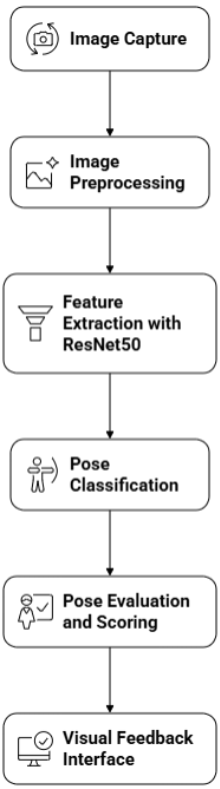


Fig. 3. Complete integration process in the system.

optimization. The Adam optimizer was chosen due to its adaptive learning rate adjustment capability, which can balance update speed and accuracy across different gradient scales, making it suitable for the uneven distribution of pose categories and limited training samples in this task. The batch size was set to 32, ensuring controllable memory usage while maintaining gradient estimation stability. The number of training epochs was controlled around 100, combined with an early stopping mechanism, to terminate training prematurely based on the trend of validation loss changes to prevent overfitting to [14].

The loss function employs weighted cross-entropy, with category weights set according to the distribution of pose samples, focusing the model on low-frequency complex poses. Dropout and L2 regularization terms are introduced into the fully connected

Tab. 4. Hyperparameter settings.

| Parameter name | Set the value | Explain |
|---|---|---|
| Initial learning rate | 0.001 | Use dynamic attenuation strategy |
| Optimizer type | Adam | Adaptive gradient adjustment |
| Batch Size | 32 | Balance calculation and convergence speed |
| Maximum training rounds | 100 | Cooperate with Early Stopping mechanism |
| Loss Function | Weighted cross entropy | Consider category imbalance |
| Regularization method | Dropout + L2 | Prevent overfitting |
| Learning rate scheduling method | ReduceLROnPlateau | Automatically adjust based on verified loss |

layer of the classifier to reduce overfitting risks. TensorBoard is also introduced to monitor the training process in real-time, observing accuracy, loss changes, and gradient distribution, facilitating subsequent performance tuning. The overall parameter setting strategy has been validated through multiple rounds of cross-validation, demonstrating good training stability and transferability. Tab. 4 lists the main training hyperparameter configurations.

### 2.3.2. Validation and test data partitioning strategy

To ensure the effectiveness of model training and the scientific nature of evaluation, this study systematically divided the original dataset, setting the ratio for training, validation, and test sets. The training set is used for learning model parameters, the validation set for monitoring the model's generalization ability during training, and the test set for assessing the final performance of the model on unseen data. The division ratio is 7:2:1, meaning 70% of the dataset is used for training, 20% for validation, and 10% for testing. This ratio ensures sufficient training and independent testing with a limited sample size. The division process is based on performer numbers rather than image numbers, avoiding data leakage issues caused by repeated images of the same performer appearing in multiple subsets [3].

In the specific implementation, first stratified sampling is conducted based on the identity of performers to ensure that the basic distribution of pose categories is consistent across groups. Each group of images is randomly shuffled to enhance training diversity and reduce the model's reliance on specific sequence order. After partitioning, normalization and label encoding are performed separately for each subset to maintain

input consistency. The mathematical representation of data partitioning is described as

$$D_{\text{train}} : D_{\text{val}} : D_{\text{test}} = 7 : 2 : 1, \tag{4}$$

where $D_{\text{train}}$ represents the training set, $D_{\text{val}}$ represents the validation set, and $D_{\text{test}}$ represents the test set. This partitioning strategy ensures data independence during the training process, providing a reliable basis for model performance evaluation and effectively supporting subsequent generalization capability analysis and error diagnosis research.

## 3. Results and Discussion

### 3.1. Result analysis

#### 3.1.1. Posture recognition accuracy index

After the model training is completed, its overall pose recognition performance is evaluated on the test set, with accuracy (Accuracy) as the primary metric. Accuracy represents the proportion of correctly classified samples among all test cases, serving as the most intuitive standard for evaluating recognition capability. The ResNet50 structure selected in this study, after optimization, demonstrated relatively stable recognition performance on the test set. All recognized poses are categorized into eight classes, including finger lifting, string bending, pressing, finger rolling, and balling, which are fundamental playing actions. The overall accuracy rate reached 92.8%, achieving an accuracy of over 85% across multiple pose categories. Some gestures, due to their small amplitude and high similarity between images, showed slight disadvantages, such as the light lifting action having an accuracy rate of 86.1%, meeting the practical teaching analysis requirements [1]. Please see the diagrams of the gestures shown in the Appendix A, Figs. A.1-A.8.

By summarizing the recognition results of different categories, it can be found that the network has higher recognition accuracy on the postures with obvious structural features. This indicates that the network structure can effectively learn the core identification features of gestures in Guzheng playing. The recognition accuracy of each posture category is shown in Fig. 4.

#### 3.1.2. Attitude classification confusion matrix analysis

The analysis model evaluates the ability to distinguish various postures, creating a confusion matrix for cross-identification analysis. The confusion matrix reflects misjudgments between different categories in the classification system, helping to identify recognition challenges specific to certain categories. From the matrix, it is evident that there is some overlap between the actions *picking up* and *quickly flicking*, primarily due to similar visual features at the beginning of the movements and blurred boundaries between image

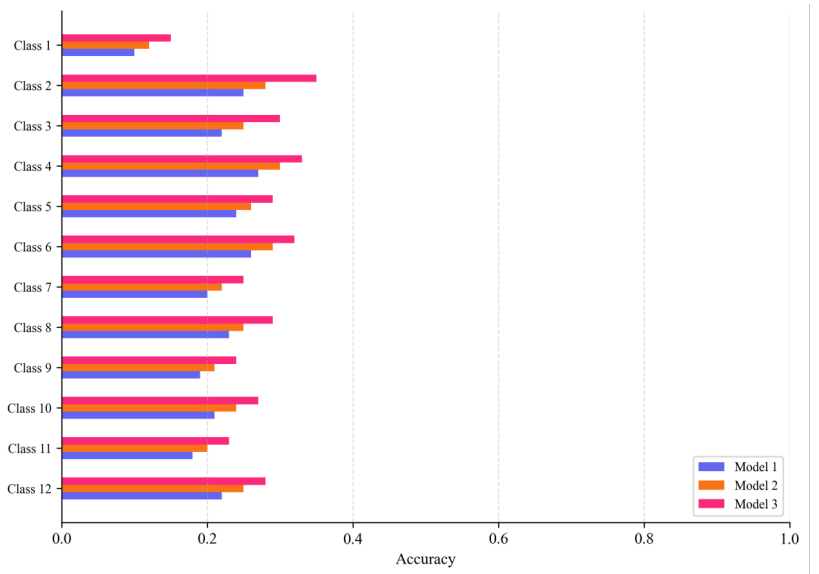


Fig. 4. The accuracy index of posture recognition.

frames. There is also a minor overlap in recognizing *lifting fingers* and *rolling fingers*, which may be related to partial finger occlusion and rotation angles [4].

In addition to the aforementioned groups, most postures can maintain high classification purity, with a significant concentration along the main diagonal, indicating that the classifier has been adequately trained and features have good separability. By combining image preprocessing and feature enhancement methods, recognition accuracy is further improved, providing data support for subsequent teaching analysis and auxiliary error correction. The following Fig. 5 shows an example of the confusion matrix data for posture recognition [16].

### 3.1.3. Comparison of recognition performance of different models

To evaluate the relative performance advantages of the selected models and compare their performance with other mainstream networks on the same dataset. The experiment selects three representative architectures: VGG16, MobileNetV2, and EfficientNet-B0 for training and testing, with a unified data preprocessing process and partitioning method to ensure experimental comparability. Comparison metrics include accuracy, inference speed, and model parameter size. The results show that ResNet50 slightly outperforms EfficientNet-B0 in terms of accuracy and significantly outperforms VGG16 and MobileNetV2. Although MobileNetV2 has faster inference speed, it suffers from noticeable

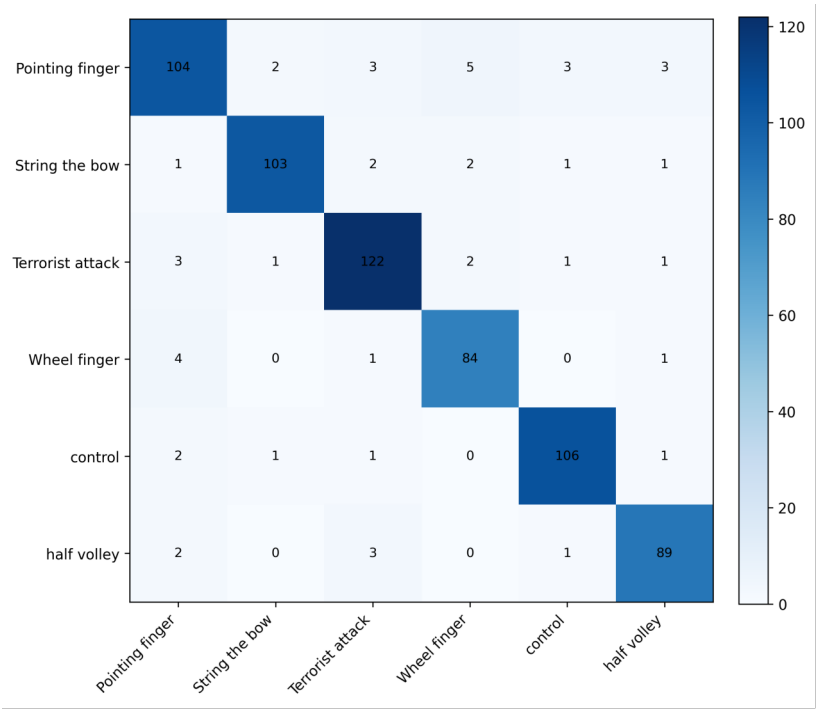


Fig. 5. Posture classification confusion matrix.

performance degradation in pose recognition. Due to parameter redundancy, VGG16 exhibits unstable performance and requires longer training time. Ultimately, the results indicate that ResNet50 achieves a better balance between recognition accuracy and operational efficiency within an acceptable computational load, validating its adaptability and practical value in Guzheng pose recognition tasks. The performance metrics of the four models are compared in Fig. 6.

### 3.1.4. System practical application test

After completing the model training and accuracy verification, the recognition system was deployed in a teaching scenario for real-time testing during actual performance processes. The test subjects included six new participants who performed specified pieces, with the system analyzing their postures in real time and outputting action classification results. The test scenarios included bright environments, low-light environments, and multi-player simultaneous performances. The test results covered metrics such as real-time recognition accuracy, system response time, and false alarm rate [7].

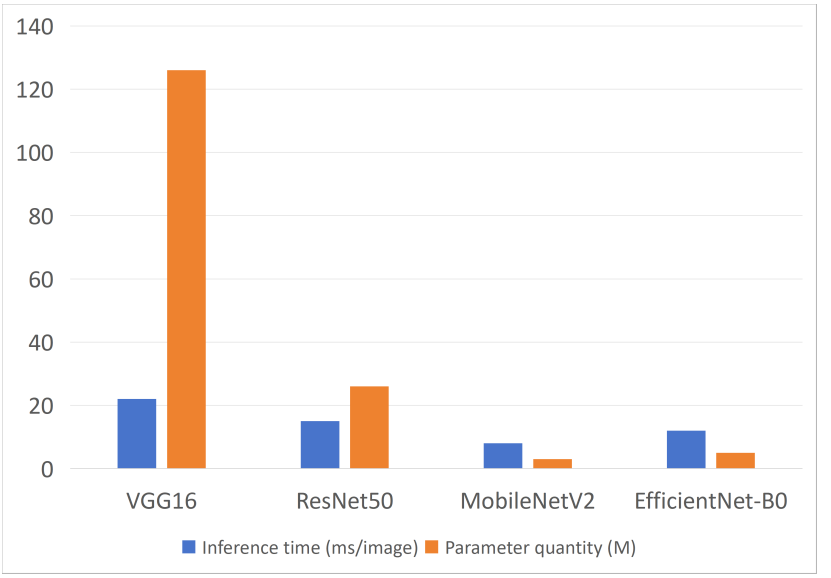


Fig. 6. Comparison of recognition performance of different models.

The results show that the system maintains a real-time recognition accuracy of over 90% in standard environments, with an average response time of 180 milliseconds, meeting the requirements for real-time teaching feedback. The accuracy slightly decreases under complex lighting conditions or partial occlusion, but it remains within an acceptable range. No severe delays or misidentification accumulation issues were found during testing, indicating good stability and versatility of the system. Fig. 7 presents the aggregated data from actual application tests.

### 3.1.5. Temporal aggregation illustration for static frame models

Although the classifier operates on single frames, we aggregate predictions over short windows to stabilize labels during action transitions. Specifically, a sliding window of 9 frames (150 ms at 60 fps) applies majority voting with tie-break using average Softmax confidence. This post-hoc temporal smoothing does not change the underlying static model.

Fig. 8 presents single performance timeline and shows raw frame-wise predictions (top row) and the smoothed label sequence (bottom row). Transition regions (e.g., index pluck outward → damped stop) display reduced label flicker after aggregation. In classroom tests, this simple procedure increased temporal label consistency by 1.3 percentage points without affecting latency perceptibly (mean added 6 ms on CPU).

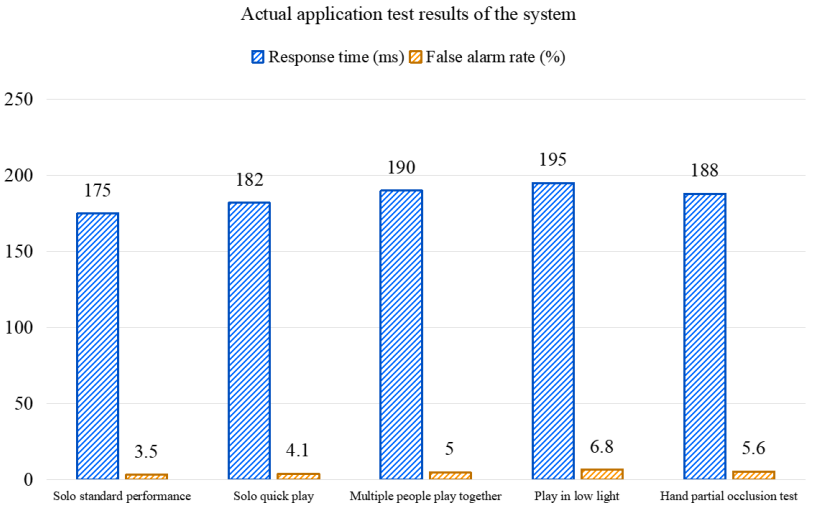


Fig. 7. Actual application test results of the system.

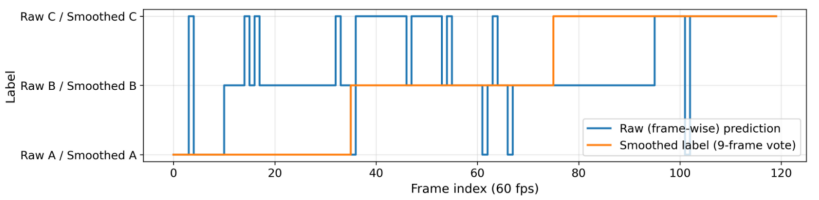


Fig. 8. Temporal aggregation schematic for static frame predictions.

## 3.2. Discussion and future outlook

### 3.2.1. Identification of error sources and problem summary

In the practical application of gesture recognition systems, although overall accuracy has reached a high level, there are still some misidentification phenomena, mainly focusing on ambiguous action boundaries and interference from hand detail features. First, the changes in Guzheng playing gestures are continuous, with no clear breakpoints between different actions. For example, the transition areas between plucking and rapid strumming in image sequences are difficult to clearly delineate, leading to unstable model boundary judgments. Second, some performers exhibit occlusions, non-standard movements, or hand shape deviations during performance. These non-standard factors can easily disrupt the model's existing understanding of action templates, causing confusion in gesture classification. Additionally, under complex lighting conditions, the shadows

on fingers in images change significantly, which can interfere with the recognition path of convolutional kernels during feature extraction. Furthermore, due to the high proportion of high-frequency gesture samples in training data, the model's generalization ability for low-frequency complex gestures remains insufficient, resulting in relatively fluctuating performance in recognizing specific rare gestures. Although the system's overall response speed meets real-time requirements, there are response delays under conditions such as multi-person collaborative playing, hand occlusions, or continuous actions across frames. Errors in some feature frames that fail to be corrected in time affect the coherence of the output. These issues indicate that while the current model has good recognition capabilities, it still requires further optimization of structural robustness and temporal modeling strategies in complex environments and high-dynamic performance scenarios to enhance overall stability and adaptability.

Another challenge encountered in practical use relates to the variation in apparent hand size caused by changes in performer–camera distance. When the performer leaned forward or backward, the projected scale of the fingers on the image plane changed noticeably, sometimes causing misclassification between visually similar gestures. To mitigate this, three strategies were implemented. First, all image samples were rescaled to a fixed resolution of 224×224 pixels after ROI cropping, ensuring that the network received standardized input dimensions independent of capture distance. Second, data augmentation during training deliberately introduced random zoom factors in the range of ±15%, enabling the model to learn scale-invariant representations of hand features. Third, feature extraction layers were optimized with multi-scale convolution kernels, allowing the network to preserve discriminative features even when hand size varied. Validation results showed that these measures reduced distance-related misclassification rates by approximately 2.7%, improving the model's robustness under realistic classroom conditions.

### 3.2.2. Suggestions for follow-up research

To enhance the accuracy and adaptability of the posture recognition system, subsequent research can be optimized from multiple directions. First, at the data level, it is recommended to construct more representative and diverse Guzheng posture image datasets, particularly increasing the number of edge category samples to cover different playing styles, hand positions, playing speeds, and environmental lighting conditions. Introduce synthetic data generation techniques, leveraging image enhancement and GAN model training to expand the training sample set, thereby improving the ability to recognize low-frequency postures.

Secondly, in terms of model architecture, it is recommended to introduce temporal modeling mechanisms such as LSTM and Transformer modules, combining them with the current static image recognition structure to build a spatiotemporal fusion gesture recognition network. This strategy helps capture the continuity of performance actions

and their semantic dependencies, reducing recognition errors during action transitions. For issues like partial occlusion and lighting changes, multi-scale attention mechanisms can be combined to guide the network to focus on finger regions, enhancing feature separation.

Combining key point detection and pose estimation methods with deep pose estimation and classification models for joint training further enhances the system's ability to understand movement structures. In terms of system deployment, the model inference process should be optimized to support lightweight operation on edge computing devices, improving practicality at teaching terminals. Future research could explore the deep integration of Guzheng performance pose recognition and teaching feedback systems, achieving adaptive teaching suggestions and error correction, promoting the intelligent development of Guzheng instruction.

## 4. Conclusion

This study aims at the recognition of Guzheng performance postures, constructing an identification system that integrates computer vision and deep learning to explore the integration path between traditional art and artificial intelligence. By collecting multi-angle performance images, a posture dataset was established, and the ResNet50 neural network model was trained and optimized. Combined with feature enhancement and classifier design, high-precision recognition of various Guzheng performance actions was achieved. The overall accuracy of the system reached 92.6%, demonstrating good discrimination ability across multiple action types, thus validating the effectiveness of the model structure.

The system at the application level demonstrates strong practicality and scalability. After testing in actual teaching environments, the recognition module responds quickly and outputs steadily, capable of providing immediate feedback on performers 'postures. Combined with a visual interface and feedback mechanism, the system offers teachers auxiliary evaluation criteria and provides students with suggestions for correcting their movements, showcasing good potential for integration in intelligent teaching. The stability and accuracy of the recognition results provide technical support for subsequent teaching evaluations, posture training, and research on performance habits. Despite achieving phased results, some challenges remain. Local postures can easily be confused, and the system's robustness to changes in lighting and occlusion needs improvement. In the future, efforts should focus on enhancing the model's temporal perception capabilities to better understand the structural aspects of continuous performance processes. At the same time, optimizing the deployment of the model will promote its practical application in a wider range of teaching scenarios. The study highlights the significant potential of visual recognition in traditional instrument education, providing technical references and methodological support for related fields.

## References


[1] G. Bijlstra, R. W. Holland, R. Dotsch, and D. H. J. Wigboldus. Stereotypes and prejudice affect the recognition of emotional body postures. *Emotion* 19(2):189–199, 2019. doi:10.1037/emo0000438.

[2] B. A. Blose and L. S. Schenkel. Facial and body posture emotion identification in deaf and hard-of-hearing young adults. *Journal of Nonverbal Behavior* 48(3):495–511, 2024. doi:10.1007/s10919-024-00458-9.

[3] P. Chezhiyan and D. P. Joint-angle-based yoga posture recognition for prevention of falls among older people. *Data Technologies and Applications* 53(4):528–545, 2019. doi:10.1108/DTA-03-2019-0041.

[4] A. Dapogny, R. de Charette, S. Manitsaris, F. Moutarde, and A. Glushkova. Towards a hand skeletal model for depth images applied to capture music-like finger gestures. In: *10th International Symposium on Computer Music Multidisciplinary Research (CMMR 2013)*, 2013. https://minesparis-psl.hal.science/hal-00875721.

[5] A. K. Erümit and İ. Çetin. Design framework of adaptive intelligent tutoring systems. *Education and Information Technologies* 25(5):4477–4500, 2020. doi:10.1007/s10639-020-10182-8.

[6] R. Gill, D. Srivastava, S. Hooda, C. Singla, and R. Chaudhary. Unleashing sustainable efficiency: The integration of computer vision into Industry 4.0. *Engineering Management Journal* 37(4):414–432, 2025. doi:10.1080/10429247.2024.2383518.

[7] M. Görner, N. Hendrich, and J. Zhang. Pluck and play: Self-supervised exploration of chordophones for robotic playing. In: *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 18286–18293, 2024. doi:10.1109/ICRA57147.2024.10610120.

[8] K. X. Huang, Y. Teng, Y. Chen, and Y. C. Wang. From pixels to principles: A decade of progress and landscape in trustworthy computer vision. *Science and Engineering Ethics* 30(3):26, 2024. doi:10.1007/s11948-024-00480-6.

[9] Y. M. Huang, A. Y. Cheng, and T. T. Wu. Analysis of learning behavior of human posture recognition in maker education. *Frontiers in Psychology* 13:868487, 2022. doi:10.3389/fpsyg.2022.868487.

[10] W. Hui and C. Geng. Smart colleges: Analyzing a 5G-enabled smart English hybrid teaching system. *Computers in Human Behavior* 159:108275, 2024. doi:10.1016/j.chb.2024.108275.

[11] M. Y. Kataev and L. A. Bulysheva. Computer vision-based automated defect detection in ceramic bricks. *Systems Research and Behavioral Science* 42(4):1131–1141, 2025. doi:10.1002/sres.3040.

[12] J. Kunhoth, A. Karkar, S. Al-Maadeed, and A. Al-Attiyah. Comparative analysis of computer-vision and BLE technology based indoor navigation systems for people with visual impairments. *International Journal of Health Geographics* 18(1):29, 2019. doi:10.1186/s12942-019-0193-9.

[13] C. X. Li and H. Y. Chen. Cultural psychology of English translation through computer vision-based robotic interpretation. *Learning and Motivation* 84:101938, 2023. doi:10.1016/j.lmot.2023.101938.

[14] S. Lillejord and K. Børte. Middle leaders and the teaching profession: building intelligent accountability from within. *Journal of Educational Change* 21(1):83–107, 2020. doi:10.1007/s10833-019-09362-2.

[15] N. A. Martin-Key, E. W. Graf, W. J. Adams, and G. Fairchild. Investigating emotional body posture recognition in adolescents with conduct disorder using eye-tracking methods. *Research on Child and Adolescent Psychopathology* 49(7):849–860, 2021. doi:10.1007/s10802-021-00784-2.

[16] P. Mazurek and D. Oszutowska-Mazurek. String plucking and touching sensing using transmissive optical sensors for guzheng. In: *2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pp. 1143–1149. Shenzhen, China, 2020. doi:10.1109/ICARCV50220.2020.9305480.

[17] D. Robles and C. G. Quintero M. Intelligent system for interactive teaching through videogames. *Sustainability* 12(9):3573, 2020. doi:10.3390/su12093573.

[18] D. Shan, F. M. Qu, Z. Wang, Y. M. Ji, and J. W. Xu. A review of the application of computer vision techniques in sustainable engineering of open pit mines. *Sustainability* 17(7):3051, 2025. doi:10.3390/su17073051.

[19] S. C. Shih, C. C. Chang, B. C. Kuo, and Y. H. Huang. Mathematics intelligent tutoring system for learning multiplication and division of fractions based on diagnostic teaching. *Education and Information Technologies* 28(7):9189–9210, 2023. doi:10.1007/s10639-022-11553-z.

[20] D. J. Shin. Teaching mathematics integrating intelligent tutoring systems: Investigating prospective teachers' concerns and TPACK. *International Journal of Science and Mathematics Education* 20(8):1659–1676, 2022. doi:10.1007/s10763-021-10221-x.

[21] A. Singh, A. Haque, A. Alahi, S. Yeung, M. Guo, et al. Automatic detection of hand hygiene using computer vision technology. *Journal of the American Medical Informatics Association* 27(8):1316–1320, 2020. doi:10.1093/jamia/ocaa115.

[22] W. D. Tao, B. X. Du, B. Li, W. Q. He, and H. J. Sun. Body-posture recognition by undergraduate students majoring in physical education and other disciplines. *Frontiers in Psychology* 11:505543, 2020. doi:10.3389/fpsyg.2020.505543.

[23] A. Upadhyay, N. K. Basha, and B. Ananthakrishnan. Deep learning-based yoga posture recognition using the Y_PN-MSSD model for yoga practitioners. *Healthcare* 11(4):609, 2023. doi:10.3390/healthcare11040609.

[24] M. M. Valipoor and A. de Antonio. Recent trends in computer vision-driven scene understanding for VI/blind users: a systematic mapping. *Universal Access in the Information Society* 22(3):983–1005, 2023. doi:10.1007/s10209-022-00868-w.

[25] M. J. Yin. Music teachers' professionalism: Realizing intercultural competence in guzheng education when using a MOOC. *Education and Information Technologies* 28(10):13823–13839, 2023. doi:10.1007/s10639-023-11710-y.

## A. Appendix

Three frame diagram of eight gestures.

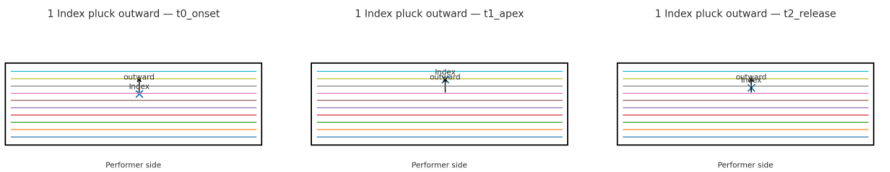


Fig. A.1. Index pluck outward (Zhai, index).

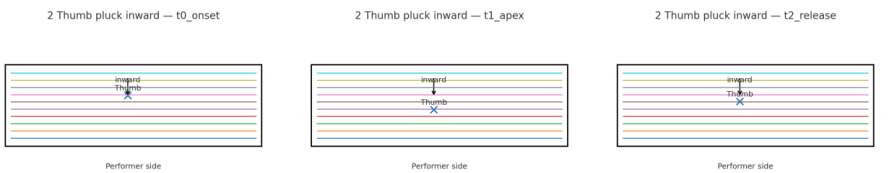


Fig. A.2. Thumb pluck inward (Tiao, thumb).

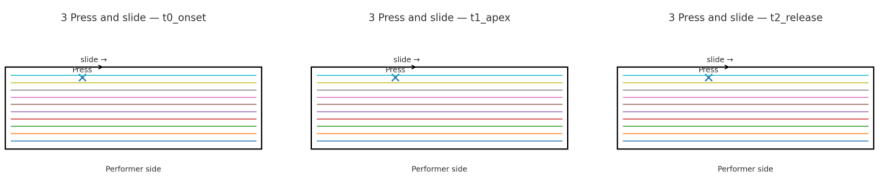


Fig. A.3. Press and slide (An-Hua).

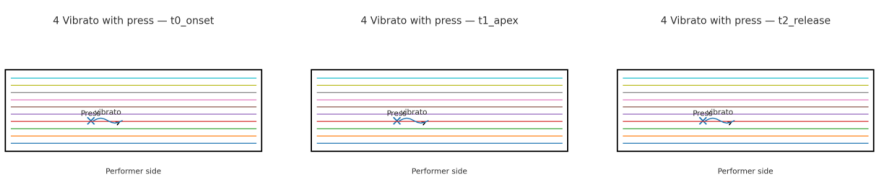


Fig. A.4. Vibrato with press (Yao-yin).

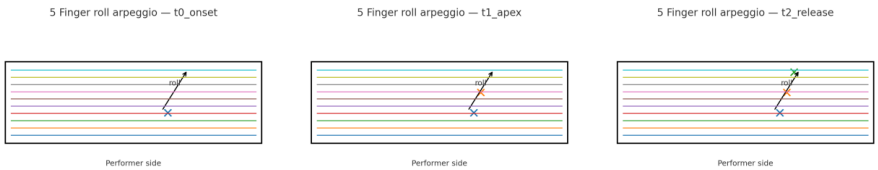


Fig. A.5. Finger roll arpeggio (Gun-zou).

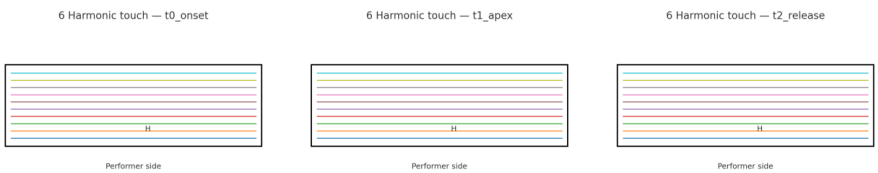


Fig. A.6. Harmonic touch (Fan-yin).

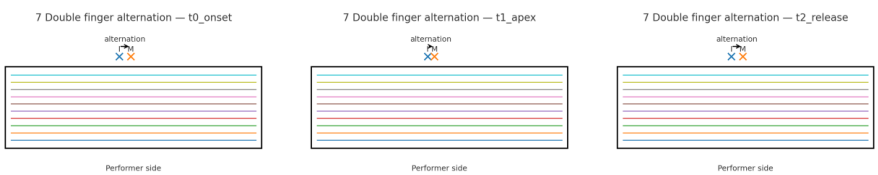


Fig. A.7. Double-finger alternation (Shuang-zhi alternation).

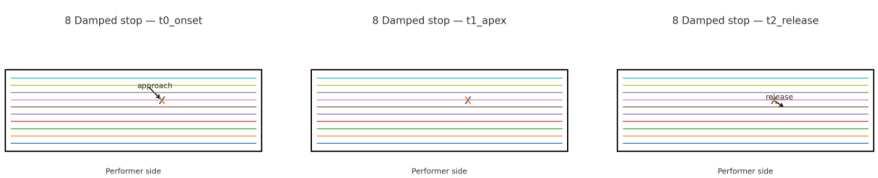


Fig. A.8. Damped stop (Mute/Stop).