ITC 3/55
Information Technology and Control
Vol. 55 / No. 3/ 2026
pp. 1227-1248
DOI 10.5755/j01.itc.55.3.43947

Gesture Interaction Design Optimization Based on Missing Information Completion and Dynamic Key Point Detection

Received 2025/12/26 | Accepted after revision 2026/06/24

HOW TO CITE: He, L., Sun, Y., Xie, X. (2026) Gesture Interaction Design Optimization Based on Missing Information Completion and Dynamic Key Point Detection. *Information Technology and Control*, 55(3), 1226-1247. https://doi.org/10.5755/j01.itc.55.3.43947

Gesture Interaction Design Optimization Based on Missing Information Completion and Dynamic Key Point Detection

Linlong He*, Yage Sun, Xiting Xie
School of Packaging Design and Art, Hunan University of Technology, Zhuzhou, 412007, China

Corresponding author: 370988642s@sina.com

With the widespread application of gesture interaction in virtual reality and intelligent driving, its design optimization has become particularly important. To address the poor accuracy in current gesture interaction design detection methods due to missing information and insufficient adaptive capabilities, a method for missing information completion based on generative adversarial networks and optical flow guided is proposed. Based on this method, the You Only Look Once version 7 algorithm and Residual Network with 50 layers are introduced to construct a detection method based on missing information completion and dynamic key point. The missing information completion method was compared. The proposed information completion method had a structural similarity, average peak signal-to-noise ratio, and running time of 97.86%, 35.76 dB, and 1.07 seconds, respectively, all of which were superior to the comparison method. Subsequently, the proposed dynamic key point detection method was compared with other methods. The results showed that the average error and correct key point detection ratio of the proposed dynamic key point detection method were 2.07px and 98.26%, which were superior to the comparison method. Finally, the proposed missing information completion method and dynamic key point detection method were applied to the in-car gesture interaction system for optimization effect analysis. The proposed method had an accuracy of 98.99% for key point detection, which was significantly better than the original method used in gesture interaction systems. The proposed method has achieved good optimization effects in gesture interaction design, and has effectiveness and practical value, providing a certain theoretical basis for optimizing gesture interaction design in related research fields.

KEYWORDS: YOLOv7; ResNet50; Generative Adversarial Network; Optical flow guided; Gesture interaction optimization; Dynamic key point detection

1. Background

Driven by artificial intelligence and machine vision, gesture interaction design has received widespread attention in human-computer interaction. Compared to traditional interactive operations such as screen touch, voice, and facial recognition, which are more convenient and efficient, key point detection is particularly important [4, 16]. However, the current detection methods used in gesture interaction design suffer from poor accuracy due to missing information and insufficient adaptability. Many experts have conducted relevant research. For example, He et al. proposed a gesture detection and tracking model using a mixture Gaussian model to optimize the accuracy of dynamic gesture recognition in human-computer interaction. The model had good detection accuracy [8]. In response to the poor accuracy in dynamic gesture recognition in human-computer interaction, Dubey proposed a recognition model based on an improved probabilistic neural network and compared it with existing methods. The results showed that the model had better performance [6]. Xiong [27] reviewed the current popular gesture recognition technologies through literature review in human-computer interaction. By carefully selecting and optimizing algorithms, the accuracy and robustness could be significantly optimized. In response to the low recognition accuracy caused by background factors such as fiber optics in radar sensor gesture recognition, Qiao et al. [21] built an airborne gesture recognition method using a novel discriminative feature. Recognition experiments were conducted on a public dataset. The accuracy reached 96.52% [21]. Khetavath et al. [11] proposed a gesture image recognition system based on heuristic optimization of manta ray foraging to solve the low efficiency in gesture recognition image processing in human-computer interaction. Comparative experimental results showed that the system had high accuracy and low error rate [11].

Completing missing information and accurately identifying hand key points are crucial for improving interaction performance in gesture interaction design. Generative Adversarial Networks (GANs) are deep learning models composed of generators and discriminators, which have advantages such as high-quality data generation and unsupervised learning. They are widely used in scenarios such as image generation and data generation [17]. You Only Look Once version 7 (YOLOv7) has fast speed and high accuracy, which is largely applied in dynamic object detection [26]. Zhang et al. [30] built a GAN-based image classification and restoration method to solve the poor performance of traditional image restoration and optimization methods in generating real textures and filling large missing areas. This method achieved significant results in real image restoration tasks [30]. Zhang et al. [28] built a GAN-based frame reconstruction model to deal with the frame loss caused by video encoding during video transmission. The model demonstrated effectiveness and superiority in two datasets [28]. Mammeri et al. [15] built a YOLOv7 to optimize the accuracy of lung nodule detection. The average detection accuracy was 81.28% [15]. To deal with the low accuracy, Nguyen et al. [18] designed a sign language recognition method using YOLOv7. The accuracy and recall of this method reached 99.4% and 99.2%, respectively [18]. Sirisha et al. [23] built an improved YOLOv7 to detect dead animals on highways. The model had better performance compared to before the improvement [23].

The above research results indicate that although there are many methods currently applied to gesture interaction design optimization, there is little research on missing information completion and dynamic key point detection, and there are problems with information loss and poor accuracy of dynamic key point detection. Therefore, to optimize gesture interaction design, a missing information completion method based on GAN and Optical Flow Guided (OFF) is designed by combining GAN with OFF. Based on the missing information completion of the video, improved YOLOv7 and Residual Network with 50 layers (ResNet50) are used to detect dynamic key points of the hand. Finally, a gesture interaction design optimization method based on missing information completion and dynamic key point detection is constructed to improve the recognition accuracy of gesture interaction. The innovation of this study is mainly presented in two aspects. Firstly, the GAN and OFF are combined to fill in missing information in gesture interaction design. Secondly, the YOLOv7 and ResNet50 are combined to optimize the real-time and accuracy of dynamic key point de-

tection in gesture interaction design. It is expected to provide a theoretical basis for optimizing gesture interaction design in related research fields.

2. Methods and Materials

2.1. Design of Missing Information Completion Method Based on GAN and OFF

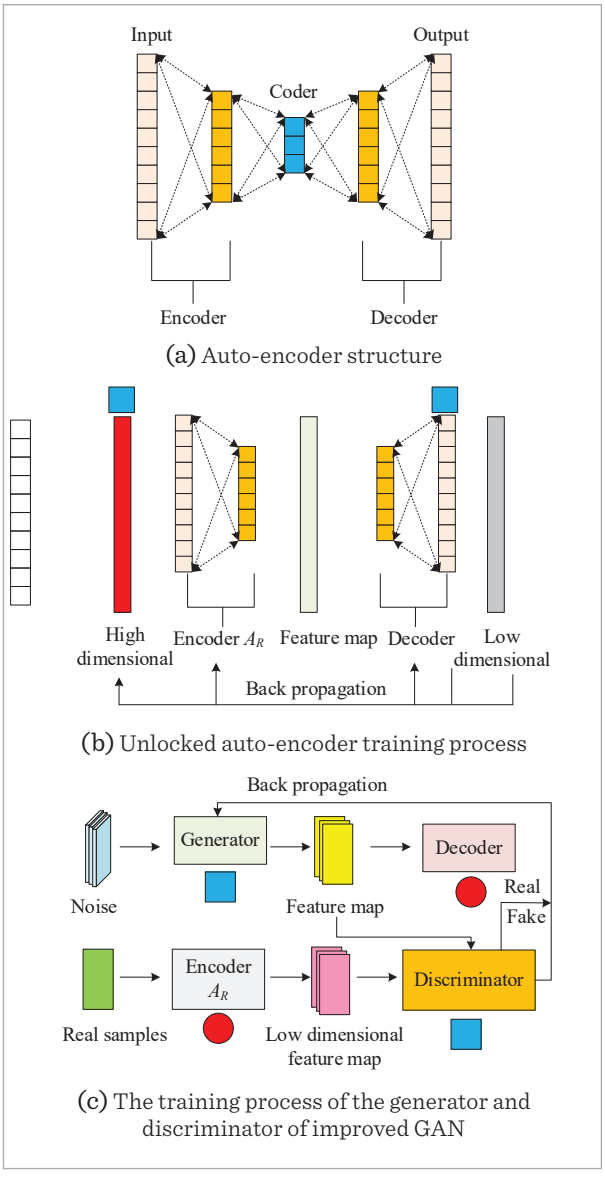
Driven by computer technology, gesture interaction has been extensively applied in virtual reality, smart homes, and in-car interaction due to its non-contact, intuitive, natural, and real-time response. However, factors such as lighting and occlusion often cause information loss when the system obtains gesture videos. Therefore, the study introduces GAN and combines it with OFF to construct a missing information completion method. GAN is a deep learning architecture mainly composed of generators and discriminators [2]. The training process of GAN first samples the noise vector from a preset random distribution and uses it as the input of the generator. The generator generates fake samples that are similar to real samples based on the input noise vector. Secondly, the discriminator receives both real samples and fake samples output by the generator. The discriminator determines the probability of a sample coming from a true sample or a false sample. Next, the loss function is obtained from the discrimination results, and the parameters of the discriminator and generator are updated through back propagation. Then, after multiple iterations, the sample distribution generated by the generator and the real sample tend to be consistent until the maximum iteration is satisfied. The GAN training is completed [3, 5]. The optimization of GAN can be represented by Equation (1).

$$\min_G \max_D V(D,G) = E_{x\sim p_x}[logD(x)+ E_{x\sim p_x}[log(1-D(G(z))] \quad (1)$$

In Equation (1), p_x is the distribution probability of the real data, D is the discriminator. E is the expectation. G is a generator. V is the objective function. z is the noise vector. x signifies a real sample. p signifies the probability. GAN uses generators and discriminators to continuously extract features from real samples and generate fake samples to achieve image information completion. However, GAN has increased failure rate as the resolution increases during information completion. The study introduces auto-encoders to improve it. Auto-encoder is an artificial neural network architecture that has strong advantages in data dimensionality reduction and feature extraction, which can compensate for the shortcomings of GAN [1]. Figure 1 presents the discriminator and generator training for auto-encoders and improved GAN.

Figure 1

Discriminator and generator training for auto-encoders and improved GAN.



In Figure 1, Figure 1(a) shows the structure of an autoencoder with a symmetric encoder-decoder architecture. Figure 1(b) depicts the pre-training process of the unlocked autoencoder. During this process, the encoder compresses the high-dimensional input into low-dimensional features, while the decoder restores the original image. At this stage, the generator and discriminator do not participate in the update temporarily. Figure 1(c) illustrates the training process of the improved generative adversarial network, in which the autoencoder is locked (parameters are frozen, represented by red circles), functioning only as a fixed feature extractor, while the noise vector is input into the generator to generate fake feature maps for adversarial training with the discriminator; the light blue rectangles represent the active training components. The red circles indicate that the training has ended and the parameters have been frozen, ensuring the stability of the training sequence by preventing the gradient backpropagation during adversarial learning from reaching the pre-trained encoder. To make the restored image as close as possible to the original image, the auto-encoder is trained using the L1 loss function L_α, as presented in Equation (2).

$$L_\alpha = \left\| Dec(A_R(I) - I \right\|_1 . \quad (2)$$

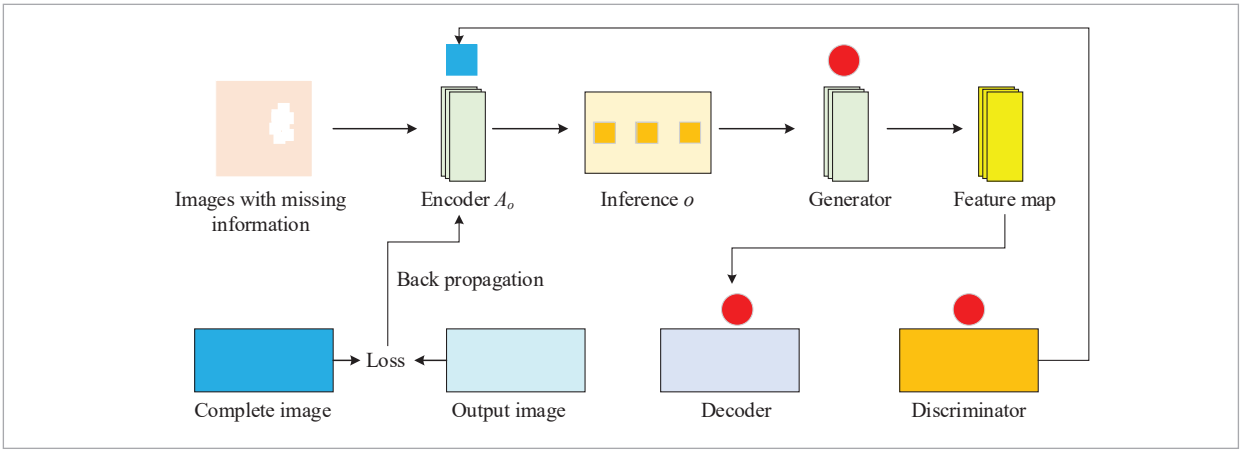
In Equation (2), *Dec* is the decoder used for dimensionality increase. I is an undamaged real image. A_R is a decoder used for dimensionality reduction. Among them, *Dec* and A_R are composed of 3 up-sampling layers and 3 down-sampling layers, respectively. The activation functions of A_R are *LeakyReLU*, *LeakyReLU*, and *Tanh*. The activation functions of *Dec* are *ReLU*, *ReLU*, and *Tanh*. The convolution kernel sizes of *Dec* and A_R are both 4×4. The number of convolution kernels for *Dec* is 128, 64, and 3, respectively. The number of convolution kernels A_R are 64, 128, and 256, respectively. Figure 1(c) shows the second part, which involves training the discriminator, generator, and locked auto-encoder. The training process first locks the encoder of the auto-encoder, which no longer participates in parameter updates and is only used as a fixed feature extractor. Secondly, the noise vector is input into the generator, and a fake feature map is generated through the generator. The encoder extracts feature maps from real images. Next, the forged feature map and the feature map of the real image are input in parallel into the discriminator. The discriminator outputs confidence intervals of [0,1] for both, and measures the true distribution. Then, depending on the output from the discriminator, the adversarial loss between the discriminator and the generator is calculated. The back propagation updates the parameters. Finally, the discriminator is trained until it is difficult to distinguish between fake feature maps and real feature maps. In addition, in this process, the discriminator adopts a multi-adversarial loss function L_D, as represented by Equation (3).

$$L_D = -E_{I\sim P_I}[logD(A_R(I))] - E_{z\sim P_z}[1 - D(A(z))] . \quad (3)$$

In Equation (3), $E_{z\sim P_z}$ represents expectations for sampling from the noise distribution, $E_{I\sim P_I}$ is the expected calculation for the real image. p_I and p_z signify the feature distributions of the real image and noise vector. To ensure training stability in the integrated GAN-autoencoder framework, a three-stage strategy with explicit generator-discriminator balance control is adopted. Stage 1: Pre-train the autoencoder for 50 epochs using L1 reconstruction loss only, with learning rate 1e-3 and Adam optimizer, to establish initial reconstruction capability and prevent mode collapse. Stage 2: Freeze autoencoder parameters and conduct adversarial training with asymmetric update rates. The discriminator is updated 5 times per generator update. The generator uses learning rate 1e-4 and the discriminator uses learning rate 4e-4. Spectral normalization is applied to all convolutional layers in the discriminator to constrain the Lipschitz constant. Gradient penalty with coefficient 10 is added to the discriminator loss to enforce the 1-Lipschitz constraint. Stage 3: Jointly fine-tune all components for 50 epochs with learning rate 1e-4. The total loss combines adversarial loss, reconstruction loss with weight 0.1, and perceptual loss with weight 0.01. Convergence criteria are monitored simultaneously: validation loss stagnation below 0.1% decrease for 10 consecutive epochs, generator-discriminator loss ratio maintained between 0.5 and 2.0, and L2 feature distance between real and generated samples below 0.01. Thus, the improved GAN training is completed. To achieve image completion, a new encoder A_o needs to be designed and trained. A_o can predict the most suitable latent encoding o for real

Figure 2

Training process of image information completion method based on improved CGN.



images with missing information. After training, it can map the missing image data to the latent space. After obtaining o, the encoding is used as input and the improved GAN trained is used to reconstruct the complete image. The loss function of A_o adopts *ReLU*, which has 6 Convolutional Layers (Conv) and 2 Fully Connected Layers (FCLs). The first Conv has a kernel size of 7×7, the second and third Convs are 5×5, and the remaining Convs are 3×3. The number of filters in the first, second, and third Convs is 64, 128, and 256. The remaining Convs have a total of 512 filters, while the first and second FCLs have 4,096 and 100 filters, respectively. The second to seventh Convs all have a batch normalization layer and use a *ReLU* loss function. Therefore, an image information completion method is constructed based on improved GAN. Figure 2 presents the training process of this completion method.

From Figure 2, the training process of this method involves first inputting damaged images with missing information into A_o to extract o. Then, o is fed into a generator to generate a complete feature map. The feature map is reconstructed into a complete output image through a decoder. During this process, training is conducted by calculating the authenticity loss and reconstruction loss. Among them, the authenticity loss is $L_p=log(1-D(G(A_o(I_M))))$, where I_M is the image with missing information. The reconstruction loss is $L_r=\|I-I_{out}\|_1$, where I_{out} is the output image. The discriminator optimizes its discriminative ability by distinguishing between real images and generated images, ultimately achieving the restoration of high-quality complete images from damaged images. The real loss is due to the frequent use of video sensors to obtain gesture data in gesture interaction, and the video frame images are dynamically variable. Therefore, the study combines OFF with an improved GAN-based information missing image completion method to construct a gesture interaction missing information completion method. This method is shown in Figure 3.

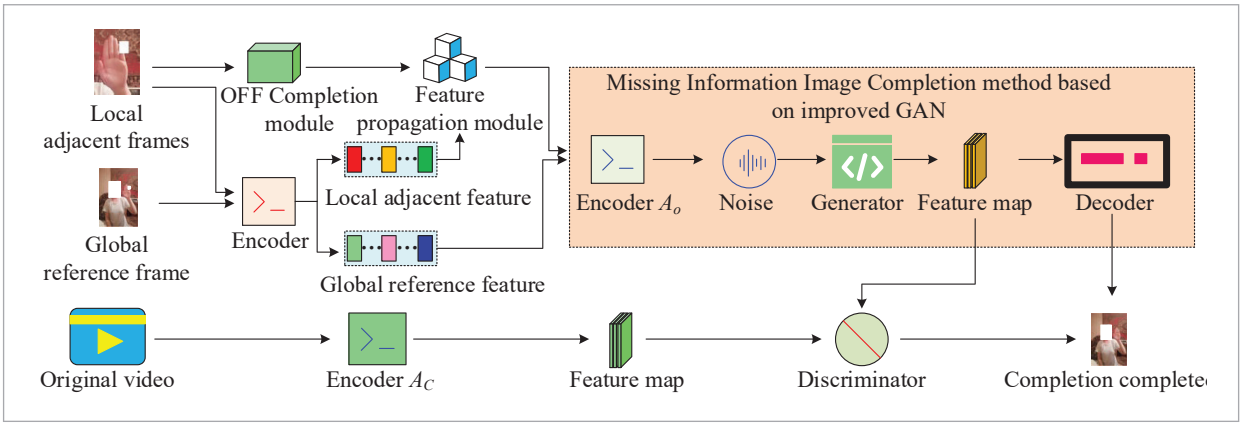
From Figure 3, the information completion method first uses a context encoder to convert the missing video frames into low resolution feature maps to improve processing efficiency. Then, the key features in the video frames are extracted through the completion module of OFF, and the feature propagation module is used to align the features between video frames. Finally, the improved GAN combines local and global feature information to effectively complete damaged video frames, thereby generating coherent and high-quality video sequences. The OFF completion information can be represented by Equation (4).

$$F_{out}^{t\to j}=T(X^t,X^j). \tag{4}$$

In Equation (4), $F_{out}^{t\to j}$ represents the output optical flow from time frame t to time frame j. T is an optical flow estimation network. X^t and X^j are adjacent sampling frames. The recovery of optical flow adopts L1 loss function. The propagation and alignment of features $C_{out}^{t\to b}$ can be represented by Equation (5).

Figure 3

The missing information completion method based on GAN and OFF (Gesture image source: https://github.com/hukenovs/hagrid).



$$C_{out}^{t\to b}=P_b(C',W(C_{out}^{(t+1)\to b},F_{out}^{t\to t+1}))\,. \tag{5}$$

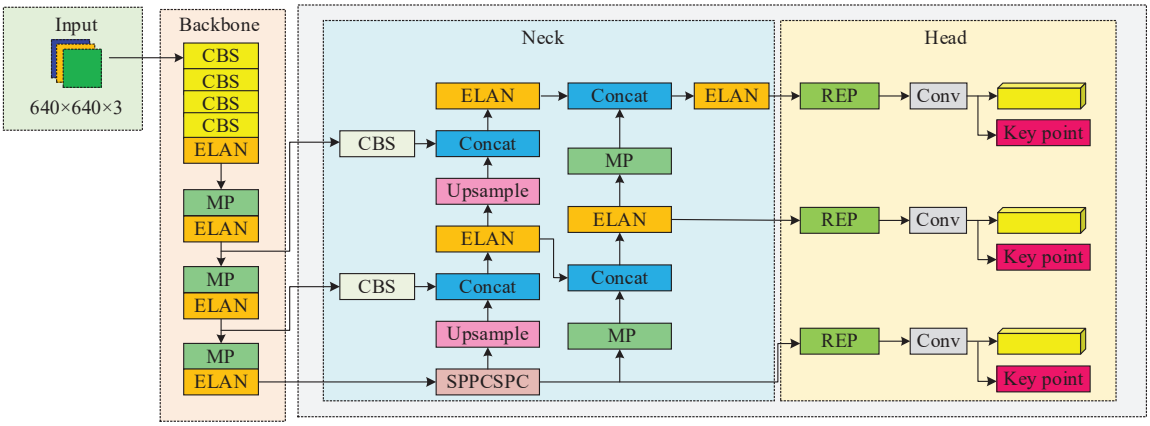
In Equation (5), P_b is the propagation function. W is the spatial distortion operation. C' is the current content feature. $C_{out}^{(t+1)\to b}$ and $F_{out}^{t\to t+1}$ are the characteristics of reverse and forward optical flow. Furthermore, as light flow refers to the pixel-level displacement produced in a continuous sequence of images due to the movement of objects. Optical flow can describe the movement direction and speed of each pixel in an image. The concept of optical flow is based on the following assumption: pixels between adjacent frames are continuous in time, and adjacent pixels are adjacent in space. To ensure the consistency and continuity of optical flow in different frames, the uniformity sensitivity threshold is defined relative to the frame diagonal length. For input resolution 640 by 640 pixels, the diagonal length is 905 pixels. The threshold is set at 2% of this diagonal length, which equals 18.1 pixels. The 2% threshold was determined empirically by evaluating the trade-off between noise suppression and motion sensitivity across the training set, where typical fingertip displacement exceeds this value while sensor jitter remains below it. A pixel is updated only when its optical flow displacement exceeds this threshold. This absolute threshold definition prevents misjudgments caused by small-scale movements or noise. The error tolerance is set to 1 pixel unit to allow for a certain degree of alignment error during the optical flow estimation process. The feature propagation operation uses bilinear interpolation over the 4-neighborhood of each pixel. Three constraints are applied to ensure stable propagation. First, the flow magnitude is clipped to a maximum of 36.2 pixels, which is twice the threshold value. Second, an occlusion mask is computed through forward-backward flow consistency checking. When the sum of forward flow and backward flow at a pixel location exceeds 1 pixel, that pixel is marked as occluded. Third, for occluded pixels, the propagated features are blended with encoder features using the occlusion mask as weights. For gesture mutation scenarios with sudden hand movement exceeding 20 pixels per frame, bidirectional optical flow pre-restoration is employed. Only flow vectors with forward-backward consistency error below 2 pixels are trusted. Inconsistent regions automatically fall back to encoder features without propagation. This error tolerance is designed to ensure that even slight alignment errors during the reconstruction of consecutive frames do not have a significant impact on the final optical flow estimation, thereby maintaining temporal consistency. When conducting gesture interaction, multi-gesture interaction may have the influence of motion discontinuity. The OFF method converts video frames with missing information into low-resolution feature maps through an encoder, then uses the completion module to extract key features, and aligns the features between video frames through the feature propagation module. When dealing with gesture mutations, this method stabilizes the optical flow by pre-restoring the for-

ward and reverse optical flows and then performing feature propagation, thereby reducing the impact of motion discontinuity. In multi-hand interaction scenarios, the OFF method can effectively handle complex optical flow information and improve the accuracy of key point detection. By combining local and global feature information, the OFF method achieves effective completion of damaged video frames and generates coherent and high-quality video sequences.

2.2. Method Based on Missing Information Completion and Dynamic Key Point Detection

Although the missing information in gesture interaction has been solved through the information completion method based on GAN and OFF, there is still low accuracy in dynamic key point detection. Therefore, an improved YOLOv7 algorithm is designed to capture the user's hand. Subsequently, ResNet50 is used to detect hand key points to optimize dynamic key point detection in gesture interaction scenarios. The feature extraction pipeline follows a strict five-stage hierarchy. The first stage is input acquisition, receiving raw video frames at resolution 640 by 640 with 3 color channels. The second stage is information completion, where the GAN and optical flow guided module restore missing information and output completed frames with identical dimensions. The third stage is hand detection, where the YOLOv7 backbone extracts hand region features and outputs feature maps at resolution 20 by 20 with 512 channels together with bounding box coordinates. The fourth stage is keypoint regression, where the cropped hand region at resolution 224 by 224 is processed by ResNet50 to output 21 keypoint coordinates. The fifth stage is temporal smoothing, where confidence-based temporal filtering and adaptive sliding window averaging are applied to sequences of keypoint coordinates to output smoothed trajectories, with optical flow constraints ensuring motion continuity across frames. The interaction rules between stages are strictly defined. The YOLOv7 detector only outputs bounding box coordinates, and the cropped hand region is resized before entering ResNet50. ResNet50 only accesses the completed image and cannot access internal features of the GAN. The temporal smoothing module only receives outputs from ResNet50, and its gradients do not propagate to the YOLOv7 detector or the GAN completion module. This decoupled design ensures that each module can be trained and optimized independently while maintaining deterministic data flow through the entire pipeline. When detecting dynamic key points, it is necessary to build an improved YOLOv7 algorithm to detect the user's hands. To optimize the detection accuracy of YOLOv7 algorithm, hand key points in the head of YOLOv7 algorithm are introduced. The accuracy of gesture detection in YOLOv7 algorithm is improved by presetting key points. After integrating the preset key points, the output tensor shape of the YOLOv7 header is adjusted from [batch_size, num_classes, height, width] to [batch_size, (num_classes + num_keypoints), height, width] to accommodate keypoint prediction. The modification of YOLOv7 with preset keypoints is technically justified as follows. For spatial priors, the 21 hand keypoints are selected based on anatomical structure and embedded into the network through a hierarchical decoding strategy where anchor boxes are initialized with bias offsets toward average hand positions, and Gaussian heatmap supervision creates spatial attention priors in early layers. Regarding output tensor changes, the detection head expands from standard object detection outputs to include 63 additional channels encoding x, y coordinates and visibility flags for 21 keypoints, processed through a dedicated keypoint regression branch with spatial dropout to prevent overfitting. For generalization impact, the preset configuration reduces the solution space from unconstrained 2D coordinates to physiologically valid hand configurations, enabling probabilistic inference of occluded keypoints through anatomical neighbor relationships, which explains the maintained accuracy under challenging conditions such as single-finger occlusion and cross-dataset testing. In addition, a branch dedicated to predicting the positions and confidence levels of key points has been added to adjust the network structure to adapt to key point detection. The improved YOLOv7 consumes completed video frames through three integrated mechanisms ensuring spatial semantics preservation and artifact suppression. (1) Spatial semantics preservation: the ResNet50 backbone extracts hierarchical features from completed frames, where shallow layers preserve edge and texture information of hand structures, and deep layers capture semantic hand configurations; multi-scale feature fusion (FPN) maintains spatial coher-

Figure 4
YOLOv7 algorithm with added hand key points.



ence across different resolutions. (2) Artifact propagation prevention: confidence-weighted masking is applied where GAN-completed regions with confidence below 0.7 are downweighted during feature extraction; optical flow consistency check further filters temporal artifacts by verifying pixel motion continuity between adjacent frames. (3) Dynamic scene adaptation: for complex environments, spatial attention mechanisms focus on hand regions suppressing background clutter; for dynamic motion, temporal feature fusion with adaptive weighting is 0.8 stabilizes keypoint detection across frames. The YOLOv7 algorithm with preset key points is shown in Figure 4.

From Figure 4, the algorithm process involves first extracting features from the hand image through the backbone network. The backbone network is mainly composed of Convolutional Block with Batch Normalization and SiLU (CBS), Efficient Layer Aggregation Network (ELAN), and Max Pooling (MP) [22]. Image features can be efficiently extracted through the backbone network. Then, the extracted features are processed through the neck. The main components of the neck include Feature Pyramid Network (FPN) and Spatial Pyramid Pooling Cross Stage Partial Connection (SPPCSPC) [7]. The ability to fuse feature maps of various scales and enhance feature expression can be improved. Finally, the accuracy of YOLOv7 algorithm in detecting users' hands is improved through the head network and pre-set hand key points. The head mainly adopts Residual Efficient Path (REP), which can improve the computational efficiency of the algorithm [19]. Due to the flexible nature of the hand, it becomes difficult to detect. Therefore, the study uses Euclidean distance to calculate the distance between the predicted key points of YOLOv7 and the preset key points. The Euclidean distance is presented in Equation (6) [29].

$$d = (\sqrt{\sum_{(\hat{x},\hat{r})} (\hat{x} - x)^2, (\hat{r} - r)^2} \,. \tag{6}$$

In Equation (6), $\hat{x}$ and $\hat{r}$ refer to the horizontal and vertical coordinates of the predicted results. d is the Euclidean distance between two coordinates. x and r refer to the horizontal and vertical coordinates of the real key points. The Euclidean distance was chosen because it has geometric interpretability and gradient stability. Its gradient can provide consistent direction guidance, unaffected by the magnitude of errors, thus avoiding oscillations in the early training stage. Combined with the wing loss, it achieves a balance of error sensitivity: small errors are penalized logarithmically to obtain more precise focusing processing, while large errors maintain a linear response to handle occlusion robustly. The loss function directly determines the direction and gradient of model optimization, and selecting the appropriate one can significantly improve accuracy, robustness, and convergence speed. To optimize the conver-

gence stationarity and learning ability of YOLOv7, the loss function is improved. Firstly, to improve its training stability, the Wing-Loss function adjusts the loss. Due to its excellent gradient characteristics, Wing Loss is used for keypoint regression. For smaller errors, the magnitude of the gradient is inversely proportional to the error, which provides higher sensitivity compared to L2 or L1 loss. For larger errors, the gradient saturates at a constant value, ensuring the stability of convergence. The analysis of the scenarios demonstrates its advantages: (1) In fingertip localization (error less than 2 pixels), the amplified gradient enables rapid convergence to sub-pixel accuracy; (2) post-occlusion recovery enables quick position correction when keypoints reappear. This gradient behavior matches hand keypoint characteristics, where sub-pixel accuracy is crucial for gesture recognition. The curvature parameter of 10 was determined based on the average displacement range of fingertips in the training set. This ensures that the transition from the logarithmic region to the linear region is consistent with the typical joint movement range of the fingers. The $Wing_{loss}(a)$ loss function is shown in Equation (7).

$$Wing_{loss}(a)=\begin{cases} wIn(1+\dfrac{|a|}{\varepsilon}), if\ |a|<w \\ |a|-H, otherwise \end{cases}. \quad (7)$$

In Equation (7), ε is the curvature of the constrained nonlinear region, and $\varepsilon \geq 0$, with a value of 2. w is a nonlinear region with a value of 10. The curvature parameter of Wing Loss was determined based on the average displacement range of fingertips in the training set. A comparative analysis was conducted with curvature values of 5, 10, and 15, where ω=10 achieved the optimal balance between small-error sensitivity and large-error stability. The keypoint loss weight and confidence loss weight were selected through experimental validation with candidate values {0.05, 0.1, 0.2} and {0.3, 0.5, 0.7} respectively, where the combination of 0.1 and 0.5 yielded the highest detection accuracy with stable training convergence. Next, to simplify the YOLOv7 algorithm structure and improve computational efficiency, the loss function *BCEWithLogitsLoss* is adopted as the key point confidence function. The *BCEWithLogitsLoss* can be represented by Equation (8).

$$BCEWithLogitsLoss=-\frac{1}{N}\sum_{i=1}^{N}- (y_i \cdot log(Sigmoid(p_i))+(1-y_i)\cdot log(1-\sigma(p_i))). \quad (8)$$

In Equation (8), N signifies the quantity of samples. y_i signifies the true label. p_i signifies the probability of the sample being predicted as a positive case. The total loss function $Loss_{N_total}$ of YOLOv7 algorithm can be represented by Equation (9).

$$Loss_{N_total}=Loss_{total}+\eta_{kpt}Loss_{kpt}+\eta_{kptv}Loss_{kptv} \quad (9)$$

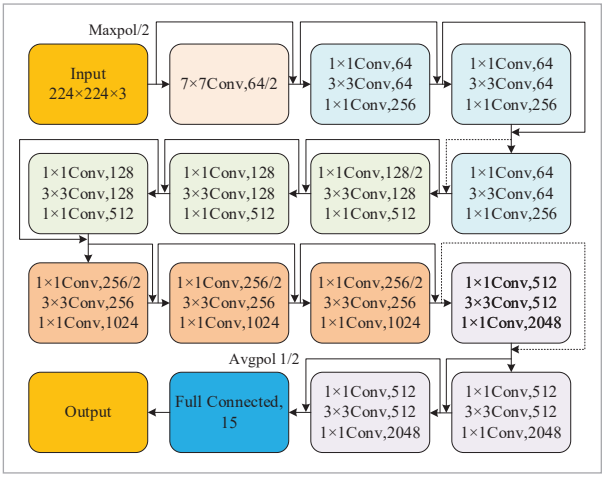
In Equation (9), η_{kpt} is the weight of the key point loss function, with a value of 0.1. $Loss_{kptv}$ is the key point confidence loss function. $Loss_{total}$ is the total loss function used for detection in the original YOLOv7. η_{kptv} is the weight of the key point confidence loss function, with a value of 0.5. Finally, to optimize the border detection performance of the YOLOv7 algorithm, improvements are made to its fitness function. The Fitness Function Weighting Strategy reshapes the optimization landscape by dynamically balancing multiple objectives during training. The fitness function integrates three components: bounding box quality, strict localization accuracy, and keypoint confidence. By assigning higher weight to boxes and lower weight to keypoints in early training, the strategy prioritizes coarse hand localization before refining keypoint precision. This staged optimization prevents premature convergence to suboptimal keypoint configurations and ensures stable multi-objective training. Under real-time constraints, the weighting strategy enables single-stage inference without iterative refinement. Prioritizing box quality in early training reduces candidate proposal overhead, decreasing NMS complexity and meeting deployment requirements while maintaining detection reliability. The weight selection follows a phased strategy. In the early training stage, higher weights are given to the quality of the boxes to ensure that the rough positioning has sufficient stability. As the training progresses, the accuracy of the key points is gradually balanced. This approach sacrifices the accuracy of the early key points in exchange for stable box convergence, avoiding situations where the key points

are precisely located but the positions are incorrect. In the case of rapid movement, this strategy prioritizes the stability of the boxes. In the case of occlusion, the confidence of the key points is given higher priority to retain the visible joint estimates. The improved fitness function *Fitness* can be represented by Equation (10).

$$Fitness = \omega_1 \cdot mAP@0.5 + \omega_2 \cdot mAP@0.5:0.95 + \omega_3 max\left(0, \frac{1}{l}\right). \quad (10)$$

In Equation (10), ω_3 is the weight of key point detection, with a value of 1. ω_2 and ω_1 are weight systems with values of 0.1 and 0.9, respectively. l is an empirical value with a value of 10. $mAP@0.5$ signifies the average accuracy that only calculates the intersection over union ratio threshold between the predicted box and the real box, at ≥ 0.5. $mAP@0.5{:}0.95$ indicates that calculates the average accuracy on 10 intersection over union ratio thresholds, and then takes the average of these 10 results to measure the overall detection performance under stricter conditions. 21 keypoints cover palm center, finger bases, tips, and intermediate joints, validated under bending, rotation, and scale variations. This configuration covers 15 degrees of freedom of hand articulation while maintaining computational efficiency. The preset keypoints are validated under motion variability including finger bending (>90°), rapid hand rotation (>180°/s), and scale variation (0.5x-2x). The anatomical prior ensures physically plausible predictions even under extreme articulation, preventing anatomically impossible configurations (e.g., finger hyperextension beyond 180°). The YOLOv7 algorithm is improved. After improving the YOLOv7 algorithm to detect the user's hand, the ResNet50 network is taken to detect hand key points. ResNet50 has strong end-to-end learning and complex feature learning capabilities, widely used in the field of bone key point detection [10, 25]. ResNet50 employs residual connections enabling effective 50-layer training. The feature transformation process is organized as follows. The first convolutional layer uses 7 by 7 kernel with 64 filters and stride 2, producing output size 112 by 112. The max pooling layer uses 3 by 3 kernel with stride 2, producing output size 56 by 56. The first residual block group contains 3 layers with 1 by 1, 3 by 3, and 1 by 1 convolutions, maintaining output size 56 by 56 with 256 channels. The second residual block group contains 4 layers with stride 2, producing output size 28 by 28 with 512 channels. The third residual block group contains 6 layers with stride 2, producing output size 14 by 14 with 1024 channels. The fourth residual block group contains 3 layers with stride 2, producing output size 7 by 7 with 2048 channels. The final 7 by 7 spatial resolution is preserved through global average pooling rather than collapsed to 1 by 1, because the spatial position information retained in 7 by 7 feature maps is critical for accurate keypoint localization. Residual connections follow the formulation of output equals transformation plus input, where the transformation is a convolutional sequence. These skip connections preserve low-level edge information from shallow layers, which is essential for fingertip localization in hand keypoint detection. ImageNet pre-training accelerates convergence. Experimental comparison shows that replacing 7 by 7 global average pooling with direct fully-connected regression increases mean per joint position error by 3.2 pixels, representing 12.7% accuracy degradation. For implementation: hierarchical features feed into a regression head with explicit layer structure. The global average pooling layer converts 7 by 7 by 2048 feature maps to 2048-dimensional vectors. The first fully-connected layer maps 2048 dimensions to 1024 dimensions with ReLU activation and dropout rate 0.5. The second fully-connected layer maps 1024 dimensions to 512 dimensions with ReLU activation. The final fully-connected layer maps 512 dimensions to 42 dimensions representing 21 keypoint coordinates, followed by Tanh activation for normalization to the range of negative one to positive one. The 42-dimensional output is reshaped to 21 keypoints multiplied by 2 coordinates. A parallel branch with identical structure outputs 21-dimensional visibility flags through sigmoid activation for occlusion detection. The 7 by 7 spatial resolution is preserved rather than collapsed to 1 by 1 because global average pooling over 7 by 7 retains spatial position information critical for keypoint localization. Coordinate denormalization converts normalized values to actual pixel positions using image width and height. The ResNet50 structure is shown in Figure 5 [13, 20].

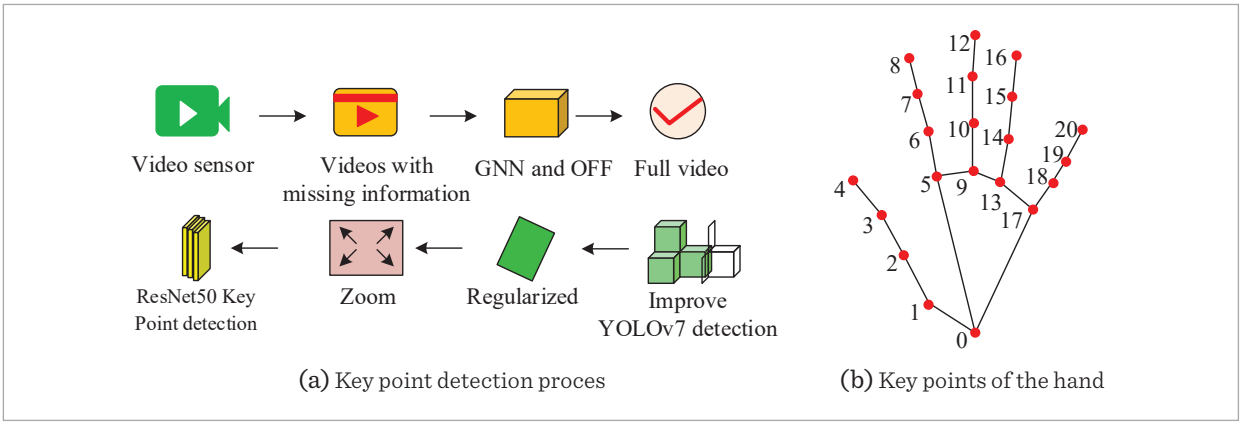
Figure 5
ResNet50 structure.



In Figure 5, 1×1Conv signifies the size of the Conv, and the number after the Conv size signifies the number of channels. From Figure 5, ResNet50 starts from an input image with a size of 224×224×3, passes through a 7×7 Conv and a MP layer, and then sequentially passes through four residual block groups, each containing different numbers of 1×1, 3×3, and 1×1 Convs. The number of output channels increases layer by layer, and outputs 15 categories of prediction results through a global average pooling layer and an FCL. Temporal jitter is suppressed through confidence-based temporal filtering, where keypoint detections with confidence below 0.7 are temporally smoothed using adjacent frame information while high-confidence detections are preserved to maintain responsiveness. Trajectory smoothing is achieved through a sliding window average with adaptive weighting based on motion velocity, where slow motions receive stronger smoothing to eliminate sensor noise and fast motions preserve original dynamics. Motion continuity is assured through optical flow constraints that flag sudden displacements exceeding 20 pixels as inconsistencies, triggering temporal interpolation from neighboring frames to maintain natural gesture flow during continuous execution. Therefore, a gesture interaction design optimization method based on missing information completion and dynamic key point detection is built. This method is shown in Figure 6.

From Figure 6(a), this method first obtains video information containing user gestures through the video sensor of the gesture interaction system. The information in this video is missing. Secondly, the complete video information is obtained through a missing information completion method based on GAN and OFF. Next, the improved YOLOv7 algorithm is used to detect the user's hand area image and preset key points. After the hand is detected, the orientation of the hand is calculated using the coordinates of the key points. Then, after calculating the orientation of the user's palm, the video frame image is scaled. Finally, the ResNet50 is input and used for forward detection of hand key points. After the detection is completed, it will be displayed in the original video frame image, and all key points can be identified by reverse operation based on the displayed key points. Figure 6(b) shows all the key points. The calculation

Figure 6
Gesture interaction design optimization method based on missing information completion and dynamic key point detection.



(a) Key point detection proces (b) Key points of the hand

of palm orientation is as follows. Firstly, the preset key points for the improved YOLOv7 algorithm are the marked key points 7 and 8 in Figure 6(b). The output angle results of the improved YOLOv7 algorithm detection show that the coordinate positions of key points 7 and 8 are $(v_1,\ k_1)$ and $(v_2,\ k_2)$, respectively. Therefore, the angle θ of their palm orientation can be represented by Equation (11).

$$\theta = \begin{cases} arctan(\frac{dk}{dv}), v_2 > v_1, k_2 > k_1 \\ arctan(-\frac{dk}{dv}) + \frac{\pi}{2}, v_2 < v_1, k_2 < k_1 \\ arctan(\frac{dk}{dv}) + \pi, v_2 < v_1, k_2 < k_1 \\ arctan(-\frac{dk}{dv}) + \frac{3\pi}{2}, v_2 > v_1, k_2 < k_1 \\ 0 \qquad\qquad , v_2 = v_1 \ or \ k_2 = k_1 \end{cases} \tag{11}$$

In Equation (11), $dv=v_2-v_1$ and $dk=k_2-k_1$. Based on the distribution of θ in the coordinate system, the rotation angle ξ can be obtained. To obtain the angle that needs to be rotated, reverse rotation can be performed to normalize the palm image. The distribution of θ and ϑ in the coordinate system is shown in Figure 7.

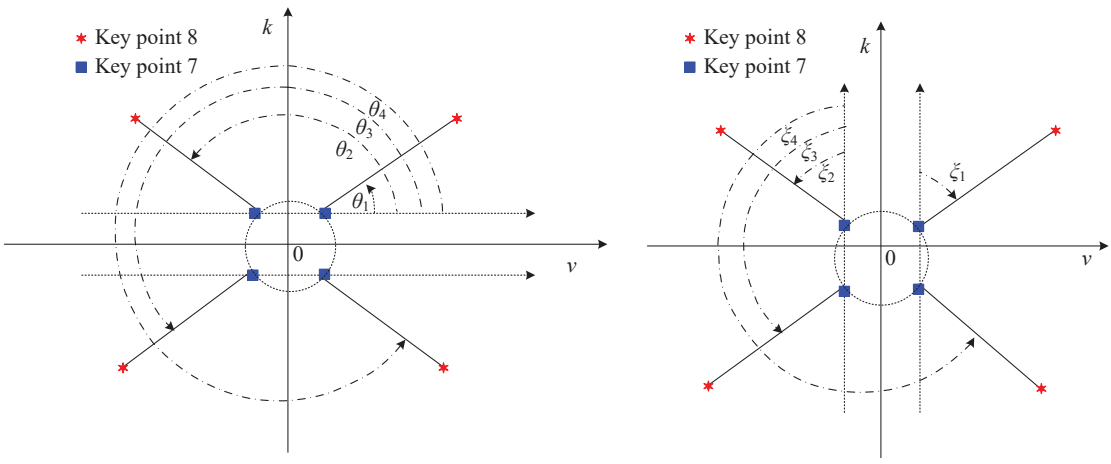
From Figure 7, the hand image is perpendicular to the coordinate system after being normalized. Calculating through coordinate system relationships can normalize hand images. The ϑ can be represented by Equation (12).

$$\xi = \begin{cases} \left|\frac{\pi}{2} - \theta\right|, v_2 \neq v_1 \ and \ k_2 \neq k_1 \\ 0, v_2 = v_1 \ or \ k_2 = k_1 \end{cases}. \tag{12}$$

Dynamic key points refer to functional points that represent the movement trajectory and temporal relationship in continuous gestures (such as the movement path and speed of the fingertip), while static key points only describe the anatomical position of the hand in a single frame. YOLOv7 and ResNet50 are responsible for frame-by-frame detection and spatial feature extraction. To ensure temporal consistency across the completion and detection modules, a lightweight temporal smoothing mechanism is applied to the keypoint coordinate sequences output by ResNet50. This mechanism combines confidence-based temporal filtering, adaptive sliding window averaging, and optical flow constraints to suppress jitter and maintain motion continuity, without interfering with the independent training of upstream modules. The input is a

Figure 7

The distribution of angles facing the palm and angles that need to be straightened.



(a) Distribution of θ in the coordinate system

(b) Distribution of ξ in the coordinate system

video frame sequence at resolution 640 by 640 with 3 color channels. The completion module receives the input sequence and outputs completed frames with identical dimensions. The detection module receives each completed frame and outputs bounding box coordinates together with confidence scores. The cropping operation extracts the hand region according to bounding box coordinates and resizes it to 224 by 224. The regression module receives the cropped hand region and outputs 21 keypoint coordinates. The temporal smoothing module receives sequences of keypoint coordinates and outputs smoothed trajectories. All transitions except the temporal smoothing module are deterministic operations with fixed mathematical rules. The temporal smoothing module is learned end-to-end but does not propagate gradients to upstream modules. This design ensures that the data flow through the entire system is transparent and reproducible.

3. Results and Analysis

3.1. Performance Analysis of Missing Information Completion Method

After constructing a missing information completion method based on GAN and OFF, the study compares its performance with other methods. Comparative methods include Homogeneous Geometry and Diffusion Model for Video Inpainting (HomoGen), Optical Flow Guided Video Frame Interpolation (VFI), and Video Super-Resolution Convolutional Neural Network (VSR-CNN). HomoGen, VFI and VSR-CNN were selected as the comparison benchmarks because they are representative in the field of video restoration and super-resolution, and have direct comparability with the proposed methods in processing video data. Although Transformer-based models and architectures based on heat map regression perform well in some tasks, they have high computational costs when processing video data and do not fully match the characteristics of the dataset. The data is sourced from the YouTube-VOS dataset and the DAVIS dataset. A total of 1,368 video clips are selected, with 70% being the training set, 20% being the validation set, and 10% being the test set. The data preprocessing adopted the image normalization standard, and Min-Max scaling was used to scale the pixel values to the range of [0, 1]. In addition, the research also implemented data augmentation strategies, including techniques such as rotation, scaling, and color transformation, to increase the diversity of the dataset and the generalization ability of the model. To address the issue of uneven distribution of gesture frequencies in the dataset, oversampling techniques and weighted loss functions were adopted. The learning rate of GAN is set to 0.001, the optimizer is Adam, and the training times are 300. The parameter settings for the auto-encoder are the same as those in the method. Comparative indicators include Structural Similarity Index (SSIM), Peak Signal-to-Noise Ratio (PSNR), Video-Based Frechet Inception Distance (VFID), Root Mean Square Error (RMSE), Mean Square Error (MSE), and running time. In addition, to ensure the reliability of the experimental results, the study conducted statistical analysis on each group of experiments. When the p value was less than 0.01, it indicated that the experimental results were reliable at the statistical 1% level. To ensure statistical reliability, all experiments are conducted with 5-fold cross-validation using different random seeds for initialization. Results report mean plus or minus standard deviation. Statistical significance is verified via paired t-test with Bonferroni correction for multiple comparisons at significance level 0.01. All reported p-values are below 0.01 after correction. To ensure the fairness and comparability of the experimental results, all experiments were conducted under a unified benchmark setting. All models were trained and tested at the same input resolution to eliminate the impact of resolution differences on model performance. All models underwent the same number of training iterations to ensure that each model had sufficient time for learning and optimization. All experiments were conducted under the same hardware configuration and software environment, including the same CPU, GPU models and operating systems, to minimize the impact of hardware differences on the experimental results. All experiments were conducted on an Intel Core i9-13900K workstation (5.8GHz, 32GB RAM) with PyTorch 1.10.0 and Python 3.7.

The comparison results of each method are shown in Table 1.

Table 1 shows the proposed method achieves the best performance across all metrics: VFID 0.031, PSNR 35.76 dB, SSIM 97.86%, runtime 1.07s, all su-

Table 1
Comparison results of each method.

Indicator	Research	HomoGen	VFI	VSR-CNN	*p* value
VFID	0.031	0.078	0.124	0.178	$P<0.01$
Average PSNR	35.76 dB	33.43 dB	34.38 dB	34.24 dB	$P<0.01$
Running time	1.07s	2.86s	3.42s	4.68s	$P<0.01$
SSIM	97.86%	91.02%	89.71%	86.54%	$P<0.01$
Average MSE	1.628	4.126	2.976	4.052	$P<0.01$
Average RMSE	0.137	0.612	0.467	0.876	$p<0.01$

Table 2
Interdependence Analysis of GAN-Autoencoder and Optical Flow Modules.

Configuration	PSNR (dB)	SSIM	Role
Full method (GAN + OFF)	40.47	0.984	Baseline
GAN only (OFF removed)	37.29	0.973	Spatial completion
OFF only (GAN removed)	33.68	0.952	Temporal propagation
Neither module (pure interpolation)	28.46	0.918	No restoration
High-res 1080p (Full method)	38.17	0.979	High-res baseline
High-res 1080p (GAN only)	29.68	0.938	Severe degradation
High-res 1080p (OFF only)	33.89	0.964	Moderate degradation

perior to HomoGen, VFI, and VSR-CNN ($p<0.01$). Training convergence is confirmed by monotonic reconstruction loss decrease and stable adversarial oscillation after epoch 80, with L2 feature distance below 0.01. In order to expand the ablation analysis and explain the interrelationship between the generative adversarial network - autoencoder and the optical flow module, the performance degradation under different module configurations was evaluated. The unbalanced performance degradation patterns that occurred in the high-resolution gesture video restoration were explained through comparative experiments. The results are shown in Table 2.

As shown in Table 2, removing GAN resulted in a degradation of 6.79 dB, and at high resolution, it was 8.49 dB, which was greater than the degradation of removing OFF of 3.18 dB, while at high resolution, removing OFF resulted in 4.28 dB. This verifies the irreplaceability of GAN content generation. The degradation gap expanded at high resolution, indicating that temporal consistency is more crucial for high-resolution restoration.

3.2. Performance Analysis of Dynamic Key Point Detection Method

After verifying the completion method performance, the study evaluates the dynamic keypoint detection method using the EgoGesture and FabDepth I datasets (3,472 samples total). Annotations are generated via Labelme and normalized to [0,1]. EgoGesture provides first-person RGB-D gesture scenarios with diverse categories, while FabDepth I supplements depth information for cross-modal validation. The dataset is split into 70% training, 20% testing, and 10% validation. Input resolution is 640×640, batch size 32, trained for 300 epochs with initial learning rate 0.01 and momentum 0.941. Comparison methods include HPEN, OpenPose, and MediaPipe, evaluated by average error, correct keypoint ratio, F1 score, and AUC across scenarios (Table 3).

Table 3
Detection results in different environments.

Scene	Method	F1 score	AUC	*p* value
Sunny day	Proposed method	98.67%	0.982	$P<0.01$
	OpenPose	89.31%	0.874	$P<0.01$
	MediaPipe	92.17%	0.825	$P<0.01$
	HPEN	88.42%	0.903	$P<0.01$
Rainy day	Proposed method	97.48%	0.962	$P<0.01$
	OpenPose	84.87%	0.857	$P<0.01$
	MediaPipe	87.69%	0.849	$P<0.01$
	HPEN	90.13%	0.886	$P<0.01$
Thick fog	Proposed method	98.32%	0.971	$P<0.01$
	OpenPose	83.67%	0.893	$P<0.01$
	MediaPipe	87.58%	0.816	$P<0.01$
	HPEN	82.99%	0.858	$P<0.01$
Occlusion	Proposed method	96.53%	0.958	$P<0.01$
	OpenPose	86.56%	0.814	$P<0.01$
	MediaPipe	79.99%	0.835	$P<0.01$
	HPEN	88.76%	0.847	$P<0.01$

Table 3 shows the proposed method achieves F1 scores above 96% across all scenarios (sunny 98.67%, rainy 97.48%, thick fog 98.32%, occlusion 96.53%) with AUC values ranging from 0.958 to 0.982, significantly outperforming HPEN, OpenPose, and MediaPipe ($p<0.01$). Ablation experiments on the 21 hand keypoints are presented in Table 4.

As can be seen from Table 4, the improved YOLOv7 algorithm significantly enhances the detection accuracy of 21 key points on the hand by integrating key point presets, fitness function weighting, and Wing loss integration. The correct key point detection rate of the basic model is 74.87%, while that of the complete model reaches 98.26%. This result verified the effectiveness of each improvement strategy, especially the Wing loss integration which played a decisive role in enhancing the detection accuracy. Furthermore, the ablation experiments indicated

Table 4
Experimental results of ablation based on the improved YOLOv7 algorithm.

Model	key-point presetting	Fitness-function weighting	wing-loss integration	Percentage of Correct Key points	*p* value
Improve YOLOv7	/	/	/	74.87%	$P<0.01$
	√	/	/	75.29%	$P<0.01$
	√	√	/	76.17%	$P<0.01$
	√	/	√	86.74%	$P<0.01$
	√	√	√	98.26%	$P<0.01$

that using any one of the strategies alone also led to an improvement over the basic model, but the combination of the three had the best effect.

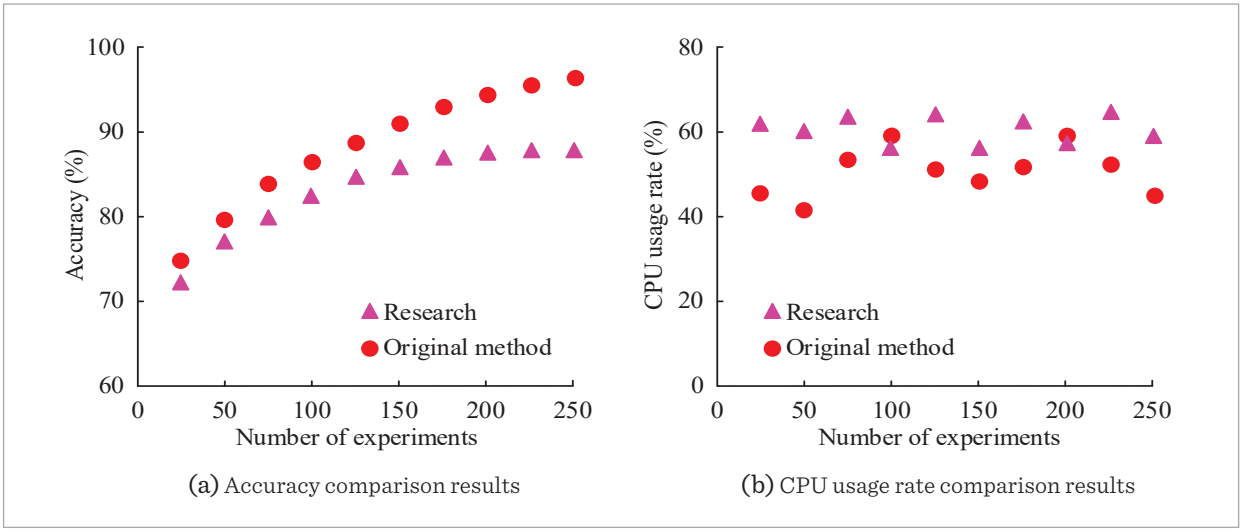
3.3. Optimization Effect Analysis

To verify the optimization effect of the missing information completion method and dynamic key point detection method on gesture interaction design, they are placed in a vehicle mounted human-computer interaction system. The comparative indicators include key point detection accuracy and CPU usage. The comparison results of key point detection accuracy and CPU usage are shown in Figure 8.

Figure 8 shows the proposed method improves keypoint detection accuracy to 98.99% from 89.27% and reduces CPU usage to 46.72% from 61.07%. This demonstrates superior precision with lower computational overhead for real-time deployment. The matching degree between the key point detection position and the actual key point position is shown in Figure 9.

Figure 8

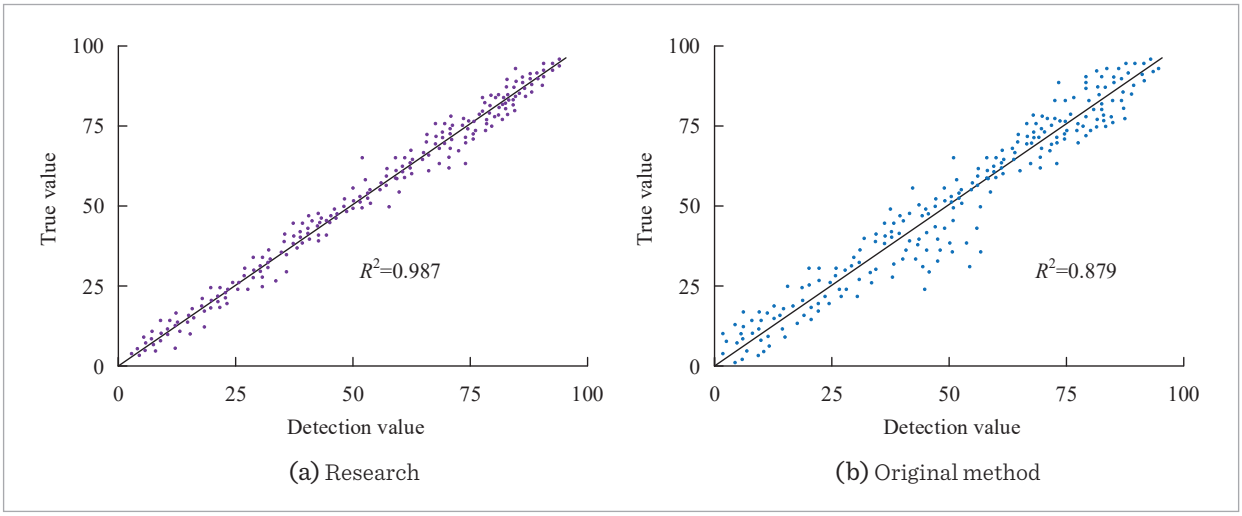
Comparison results of key point detection accuracy and CPU usage.



(a) Accuracy comparison results (b) CPU usage rate comparison results

Figure 9

Matching degree results of key point detection locations and actual key point locations.



(a) Research (b) Original method

From Figure 9(a), after adopting the method proposed in the study, the system had the highest matching between the key point position and the actual position, and the coefficient of determination R^2 was 0.987. From Figure 9(b), the matching degree between the key point positions detected by the original system and the actual positions was low, and the R^2 was 0.879, lower than that of the proposed method. The high matching between the key point detection position and the actual key point position indicates that the detection results are highly consistent with the true values, and have high accuracy and reliability. The research method improves the detection performance. The proposed method has good practical value and effectiveness. In order to verify the performance of the proposed key point detection method using the ResNet50 backbone network, the study will replace the MobileNet-V3, HRNet and Vision Transformers models for performance comparison and analysis. The analysis results are shown in Table 5.

As can be seen from Table 5, the improved YOLOv7 algorithm using ResNet50 as the backbone network model performs best in VFID and average PSNR indicators, which are 0.031 and 35.76dB respectively, and the running time is 1.07s, demonstrating good accuracy and reasonable computational efficiency. In contrast, although the running time of MobileNet-V3 was 1.28 seconds, the VFID and PSNR were 0.062 and 34.47dB respectively. HRNet and Vision Transformers have relatively long running times, which are 1.37 seconds and 2.47 seconds respectively, and they are also inferior to ResNet50 in VFID and PSNR. These results indicate that when the ResNet50 network is adopted, the performance of gesture key point detection is superior to that of the comparison methods. To further verify the impact of the improved method on the system, a comparative experimental analysis was conducted on it in terms of FLOPs, parameter quantity, memory usage and energy consumption. The experimental results are shown in Table 6.

It can be seen from Table 6 that after adopting the improved gesture key point detection method, the FLOPs decreased from 5.38G to 4.12G, a reduction of 23.47%. The number of parameters decreased from 29.89M to 25.56M, a reduction of 14.48%. The inference delay was reduced from 148ms to 97.2ms, a decrease of 34.46%. Memory usage decreased from 1879MB to 981MB, a reduction of 48.02%. and the average system power drops from 3.84 W to 2.51 W, saving 34.6 % energy under continuous processing on the battery-powered Jetson Nano platform. The

Table 5

Performance comparison and analysis results of key point detection using ResNet50.

Method	VFID	Average PSNR	Running time	*p* value
Improve YOLOv7+ResNet50	0.031	35.76dB	1.07s	$P<0.01$
Improve YOLOv7+MobileNet-V3	0.062	34.47dB	1.28s	$P<0.01$
Improve YOLOv7+HRNet	0.058	34.26dB	1.37s	$P<0.01$
Improve YOLOv7+Vision Transformers	0.068	33.14dB	2.47s	$P<0.01$

Table 6

Comparison results of FLOPs, parameter quantity, memory usage and energy consumption.

Indicator	Original system	Improvement method	*p* value
FLOPs (G)	5.38	4.12	$P<0.01$
Number of parameters (M)	29.89	25.56	$P<0.01$
Inference latency (ms)	148	97.2	$P<0.01$
Average power (W)	3.84	2.51	$P<0.01$
Memory Footprint (MB)	1879	981	$P<0.01$

Table 7
Results of Sensitivity Analysis.

Error condition	VFID	MPJPE (mm)	Accuracy rate (%)	*p* value
Benchmark	0.031	9.2 mm	98.99	$P<0.01$
The optical flow is 10% missing	0.033	10.1 mm	97.83	$P<0.01$
The optical flow is 20% missing	0.035	11.4 mm	96.69	$P<0.01$
KP σ = 1 px	0.036	9.7 mm	98.09	$P<0.01$
KP σ = 2 px	0.041	10.9 mm	96.82	$P<0.01$
KP σ = 3 px	0.045	12.5 mm	95.39	$P<0.01$

Table 8
Comparison results of stability and jitter reduction effects.

Indicator	E2FGVI	ProPainter	The proposed detection method	*p* value
Time consistency	0.59	0.69	0.88	$P<0.01$
The continuity of continuous motion	0.54	0.64	0.69	$P<0.01$
Jitter reduction effect	0.33	0.44	0.54	$P<0.01$

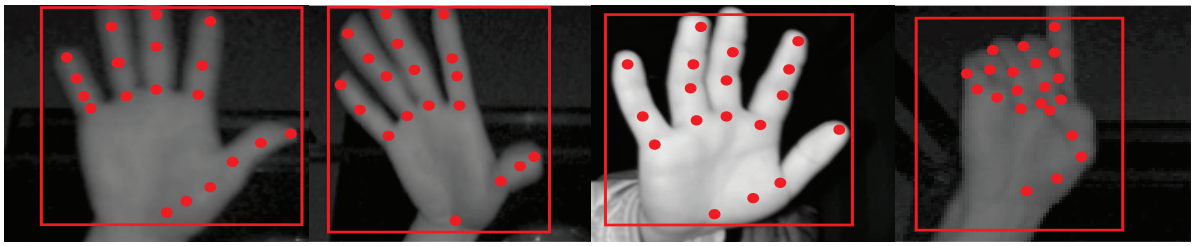
above results indicate that the method proposed in the research improves the computing efficiency and resource utilization efficiency of the system. To verify the performance of detection for the decline in optical flow estimation accuracy and the disturbance in key point positioning, the sensitivity analysis of the detection method for adding noise at key points and different optical flow accuracies was studied. The analysis results are shown in Table 7.

In Table 7, KP σ represents Key-Point perturbation with Gaussian noise standard deviation σ and Mean Per Joint Position Error. As can be seen from Table 11, under the conditions of 20% optical flow absence or KPσ=3px disturbance, the VFID only rises to 0.045, the MPJPE increases to 12.5mm, and the accuracy rate still remains above 95%, with a performance degradation of less than 3.6 percentage points, indicating that the system has good robustness against optical flow errors and key point drift. To verify the milestone stability and jitter reduction effectiveness of the proposed method in continuous motion frame-to-frame, it was placed in a vehicle-mounted system and a performance comparison analysis was conducted with the E2FGVI and ProPainter methods. The analysis results are shown in Table 8.

As can be seen from Table 8, the proposed detection method scores 0.88 in terms of time consistency, which is significantly higher than the 0.59 of E2FGVI and the 0.69 of ProPainter. The proposed detection method scored 0.69 in the continuity index of continuous motion, which was higher than that of the comparison method. In addition, in terms of the jitter reduction effect index, the proposed method also leads, scoring 0.54, while E2FGVI and ProPainter score 0.33 and 0.44 respectively. The above results indicate that the proposed method has significant advantages in improving the stability of gesture recognition and reducing jitter, which is conducive to enhancing the overall performance of the interaction system. To verify the detection effect of the method proposed in the study on continuous movements, the study placed it under different lighting conditions for detection. The correct and incorrect detection results are shown in Figure 10.

It can be seen from Figure 10(a) to Figure 10(c) that the method proposed in the research can identify the key points of gestures quite well. However, there are still recognition errors when facing the complex actions shown in Figure 10(d).

Figure 10
Actual detection effect.



(a) Action 1 (b) Action 2 (c) Action 3 (d) False results

4. Discussion and Interpretation

This study compared the performance of the missing information completion method with dynamic key point detection methods based on improved YOLOv7 and ResNet50. Two methods were applied to the in-car gesture interaction system for optimization effect analysis. In the comparative analysis of experimental results, the proposed missing information completion method was found to be effective on PSNR value, VFID, runtime, and SSIM value, demonstrating excellent performance. The average PSNR values of the information completion method, HomoGen, VFI, and VSR-CNN methods were 35.76 dB, 33.43 dB, 34.38 dB, and 34.24 dB, respectively. The information completion method had the highest average PSNR. This result indicates that the GAN improves the quality of video information completion in gesture interaction. This result coincides with the relevant research findings drawn by Hu et al. [9]. In the comparative analysis of VFID and SSIM, the VFID values of the proposed information completion method, HomoGen, VFI, and VSR-CNN methods were 0.031, 0.078, 0.124, and 0.178, respectively. The SSIM values were 97.86%, 91.02%, 89.71%, and 86.54%, respectively. The information completion method proposed in the study has better VFID and SSIM values than those of comparison methods. This result indicates that the combination of auto-encoder, GAN, and OFF improves the accuracy and quality of video frame restoration in video information completion methods. This result is matched with the relevant results obtained by Li et al. [14]. In addition, the information completion method had a running time, MSE value, and RMSE value of 1.07s, 1.628, and 0.137, which were all superior to the comparison method. This finding further validates the superior performance of the information completion method.

Subsequently, the proposed dynamic key point detection method was compared with other methods, taking average error, correct key point detection ratio, and detection performance in different scenarios as indicators. The average errors of the proposed dynamic key point detection method, HPEN, OpenPose, and MediaPipe detection methods were 2.07px, 3.86px, 4.12px, and 4.98, respectively. The correct key point detection rates were 98.26%, 84.37%, 88.62%, and 89.34%, respectively. The dynamic key point detection method had the lowest average error and the highest proportion of correct key point detection, better than those of comparison methods. The improved YOLOv7 and the ResNet50 have enhanced the accuracy of the detection method. This result is consistent with the relevant research findings drawn by Kumar A and Dhanalakshmi R in 2024 [12]. In different scene detection effects, the proposed detection method achieved F1 scores of 98.67%, 97.48%, 98.32%, and 96.53% in sunny, rainy, thick fog, and occlusion conditions, respectively. The AUC values were 0.982, 0.962, 0.971, and 0.958, respectively. Among them, the dynamic key point detection method had the highest F1 score and AUC. The proposed missing information completion method improves the accuracy of the detection method for dynamic key detection in gesture interaction systems. This result is matched with the finding drawn by Song et al. in 2024 [24].

Finally, the proposed missing information completion method and dynamic key point detection method are applied to the in-vehicle gesture interaction system. The optimization effect is analyzed. The key point detection accuracy of the research method and the original method was 98.99% and 89.27%, respectively. The average CPU usage rates were 46.72% and 61.07%, respectively. Among them, the accuracy and average CPU usage of the system key point detection method were much higher than those of the original method. In addition, the system using the proposed research method has the highest matching between the key point positions and the actual positions, with a determination coefficient R^2 of 0.987. The research method improves the gesture interaction system. From the above results, it can be seen that the difference between this research and previous studies lies in that the proposed method not only adopts GAN for image processing and optimization, but also integrates OFF technology to improve the temporal consistency between video frames. In addition, system-level evaluation methodology has been clarified. The computational efficiency is measured by FLOPs reduction through optimized optical flow guidance. Inference latency is maintained at satisfactory levels per frame for real-time requirements. Memory optimization is achieved through feature map reuse and fixed-point quantization. Under operational constraints of real-time processing and limited memory resources, the method preserves high detection accuracy, validating effectiveness for real-time vehicle deployment. To address the end-to-end performance validation, the error propagation between modules is analyzed. The information completion stage (GAN+OFF) serves as the foundation for subsequent detection; degradation in PSNR (e.g., from 36.09 dB to 29.97 dB under fast motion, Table 4) directly constrains the upper bound of keypoint detection accuracy. Ablation studies (Table 6) confirm that removal of completion modules causes error propagation to the detection stage, resulting in system failure. This establishes the causal dependency: completion quality determines detection reliability, validating the end-to-end optimization claims. Compared with the GAN models mainly focusing on image restoration in reference [30], the method in this study enhances spatio-temporal consistency by combining optical flow information, which is not covered by the video error hiding method in reference [28]. Furthermore, this study particularly emphasizes the real-time detection of key points of dynamic gestures, which was not fully reflected in the static image defogging method in reference [14]. Through the improved YOLOv7 and ResNet50, this study achieved more accurate dynamic keypoint detection. This method has effectiveness and practical value. The limitation of this study is that gesture interaction in actual environments is also affected by complex working conditions such as high-speed driving and low light at night. Future work will combine image enhancement, multi-modal fusion, edge computing optimization and other technologies to further optimize the robustness and real-time performance in extreme scenes. Future work will extend the model to three directions: (1) gesture recognition via Transformer-based temporal modeling of 21 keypoint sequences; (2) multi-hand/multi-user interaction with N-instance detection and lightweight identity association; (3) 3D pose estimation via stereo depth refinement (current root-relative 3D MPJPE: 9.4mm). These extensions will be integrated into an edge computing framework to maintain real-time 30fps performance under extreme vehicle conditions.

5. Summary

A missing information completion method was proposed to address the poor accuracy of current gesture interaction design detection methods. A dynamic key point detection method based on improved YOLOv7 and ResNet50 was constructed. A performance comparison analysis was conducted between the proposed missing information completion method and other methods. This method demonstrated superior performance on PSNR, SSIM, and running time, all of which were superior to the comparison method. Subsequently, the proposed dynamic key point detection method was compared with other methods. The method outperforms the comparison methods on average error, correct key point detection ratio, and detection performance in different scenarios. The research method is used in-vehicle system. It is found that the research method outperforms the comparison method on key point detection accuracy and CPU usage, which helps to improve the accuracy of gesture interaction design.

6. Fundings

The research is supported by the Hunan Provincial Education Science "14th Five-Year Plan" 2021 Project, Research on the application of virtual reality technology in practical teaching of design majors, ND213075; key Scientific Research Project of the Education Department of Hunan Province, Research on innovative design of Liling ceramics driven by artificial intelligence technology, 24A0424.

References

1. Berahmand, K., Bahadori, S., Abadeh, M. N., Li, Y., Xu, Y. Sdac-DA: Semi-Supervised Deep Attributed Clustering Using Dual Autoencoder. IEEE Transactions on Knowledge and Data Engineering, 2024, 36(11), 6989-7002. https://doi.org/10.1109/TKDE.2024.3389049
2. Buffolo, M., Favero, D., Marcuzzi, A., De Santi, C., Meneghesso, G., Zanoni, E., Meneghini, M. Review and Outlook on GaN and SiC Power Devices: Industrial State-of-the-Art, Applications, and Perspectives. IEEE Transactions on Electron Devices, 2024, 71(3), 1344-1355. https://doi.org/10.1109/TED.2023.3346369
3. Cao, Z., Liu, K., Jin, X., Ning, L., Yin, L., Lu, F. STAGE: A Spatiotemporal-Knowledge Enhanced Multi-Task Generative Adversarial Network (GAN) for Trajectory Generation. International Journal of Geographical Information Science, 2025, 39(5), 1100-1127. https://doi.org/10.1080/13658816.2024.2381146
4. Cheng, Y., Wu, Z., Xiao, R. Exploring Methods to Optimize Gesture Elicitation Studies: A Systematic Literature Review. IEEE Access, 2024, 12(4), 64958-64979. https://doi.org/10.1109/ACCESS.2024.3387269
5. Dash, A., Ye, J., Wang, G. A Review of Generative Adversarial Networks (GANs) and Its Applications in a Wide Variety of Disciplines: From Medical to Remote Sensing. IEEE Access, 2023, 12(1), 18330-18357. https://doi.org/10.1109/ACCESS.2023.3346273
6. Dubey, A. K. Enhanced Hand-Gesture Recognition by Improved Beetle Swarm Optimized Probabilistic Neural Network for Human-Computer Interaction. Journal of Ambient Intelligence and Humanized Computing, 2023, 14(9), 12035-12048. https://doi.org/10.1007/s12652-022-03753-9
7. Guo, J., Yang, Y., Lin, X., Memon, M. S., Liu, W., Zhang, M., Sun, E. Revolutionizing Agriculture: Real-Time Ripe Tomato Detection with the Enhanced Tomato-YOLOv7 System. IEEE Access, 2023, 11(1), 133086-133098. https://doi.org/10.1109/ACCESS.2023.3336562
8. He, D., Yang, Y., Wu, R. Design of Human-Computer Interaction Gesture Tracking Model Based on Improved PSO and KCF Algorithms. IEEE Access, 2024, 12, 39913-39925. https://doi.org/10.1109/ACCESS.2024.3375351
9. Hu, K., Liu, Y., Liu, R., Lu, W., Yu, G., Fu, B. Enhancing Quality of Pose-Varied Face Restoration with Local Weak Feature Sensing and GAN Prior. Neural Computing and Applications, 2024, 36(1), 399-412. https://doi.org/10.1007/s00521-023-09013-x
10. Khalifa, N. E., Wang, W., Mawgoud, A. A., Hang, Y. D. COECG-ResNet-GWO-SVM: An Optimized COVID-19 Electrocardiography Classification Model Based on ResNet50, Grey Wolf Optimization and Support Vector Machine. Multimedia Tools and Applications, 2025, 84(17), 18305-18325. https://doi.org/10.1007/s11042-024-19733-4
11. Khetavath, S., Sendhilkumar, N. C., Mukunthan, P., Jana, S., Malliga, L., Gopalakrishnan, S., Farhaoui, Y. An Intelligent Heuristic Manta-Ray Foraging Optimization and Adaptive Extreme Learning Machine for Hand Gesture Image Recognition. Big Data Mining and Analytics, 2023, 6(3), 321-335. https://doi.org/10.26599/BDMA.2022.9020036
12. Kumar, A., Dhanalakshmi, R. EYE-YOLO: A Multi-Spatial Pyramid Pooling and Focal-EIOU Loss Inspired Tiny YOLOv7 for Fundus Eye Disease Detection. International Journal of Intelligent Computing and Cybernetics, 2024, 17(3), 503-522. https://doi.org/10.1108/IJICC-02-2024-0077
13. Kumar, M., Arora, A., Deb, A., Yadav, A. L. Deep Learning for Accurate Plant Disease Classification Using ResNet50: A Comprehensive Approach. Proceedings of the International Conference on Computational Intelligence and Communication Applications (ICCICA), 2024, 125-130. https://doi.org/10.1109/ICCICA60014.2024.10584860

14. Li, S., Liu, F., Wei, J. Two-Stage Underwater Image Restoration Based on GAN and Optical Model. Signal, Image and Video Processing, 2024, 18(1), 379-388. https://doi.org/10.1007/s11760-023-02718-5

15. Mammeri, S., Amroune, M., Haouam, M. Y., Bendib, I., Silva, A. Corrêa. Early Detection and Diagnosis of Lung Cancer Using YOLO v7 and Transfer Learning. Multimedia Tools and Applications, 2024, 83(10), 30965-30980. https://doi.org/10.1007/s11042-023-16864-y

16. Mokayed, H., Quan, T. Z., Alkhaled, L., Sivakumar, V. Real-Time Human Detection and Counting System Using Deep Learning Computer Vision Techniques. Artificial Intelligence Applications, 2023, 1(4), 221-229. https://doi.org/10.47852/bonviewAIA2202391

17. Nayak, A., Venugopala, P. S., Ashwini, B. A Systematic Review on Generative Adversarial Network (GAN): Challenges and Future Directions. Archives of Computational Methods in Engineering, 2024, 31(8), 4739-4772. https://doi.org/10.1007/s11831-024-10119-1

18. Nguyen, T. H., Ngo, B. V., Nguyen, T. L., Vu, C. C. Finger Gesture Recognition by MediaPipe Algorithm and Advanced YOLOv7 Network for Deaf People. International Journal of Biomedical Engineering and Technology, 2025, 48(1), 55-74. https://doi.org/10.1504/IJBET.2025.146402

19. Olorunshola, O. E., Irhebhude, M. E., Evwiekpaefe, A. E. A Comparative Study of YOLOv5 and YOLOv7 Object Detection Algorithms. Journal of Computing and Social Informatics, 2023, 2(1), 1-12. https://doi.org/10.33736/jcsi.5070.2023

20. Ou, Y., Yu, W. H., Un, K. F., Chan, C. H., Zhu, Y. A 119.64 GOPs/W FPGA-Based ResNet50 Mixed-Precision Accelerator Using the Dynamic DSP Packing. IEEE Transactions on Circuits and Systems II: Express Briefs, 2024, 71(5), 2554-2558. https://doi.org/10.1109/TCSII.2024.3377356

21. Qiao, L., Li, Z., Xiao, B., Shu, Y., Li, W., Gao, X. Gesture-ProxylessNAS: A Lightweight Network for Mid-Air Gesture Recognition Based on UWB Radar. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2023, 16(1), 5144-5153. https://doi.org/10.1109/JSTARS.2023.3274830

22. Saputri, H. A., Avrillio, M., Christofer, L., Simanjaya, V., Alam, I. N. Implementation of YOLO v7 Algorithm in Estimating Traffic Flow in Malang. Procedia Computer Science, 2024, 245, 117-126. https://doi.org/10.1016/j.procs.2024.10.235

23. Sirisha, U., Chandana, B. S., Harikiran, J. Nam-YOLOv7: An Improved YOLOv7 Based on Attention Model for Animal Death Detection. Traitement du Signal, 2023, 40(2), 783-789. https://doi.org/10.18280/ts.400239

24. Song, Y., Guo, J., Wu, G., Ma, F., Li, F. Automatic Recognition of Landslides Based on YOLOv7 and Attention Mechanism. Journal of Multimodal Sciences, 2024, 21(8), 2681-2695. https://doi.org/10.1007/s11629-024-8669-x

25. Wan, J., Li, B., Wang, K., Teng, X., Wang, T., Mao, B. An Improved ResNet50 for Environment Image Classification. Procedia Computer Science, 2024, 242(1), 1000-1007. https://doi.org/10.1016/j.procs.2024.08.246

26. Wang, Y., Wang, H., Xin, Z. Efficient Detection Model of Steel Strip Surface Defects Based on YOLO-V7. IEEE Access, 2022, 10, 133936-133944. https://doi.org/10.1109/ACCESS.2022.3230894

27. Xiong, Y. Research on Gesture Recognition Technology Based on Machine Learning. Applied and Computational Engineering, 2024, 104(1), 116-121. https://doi.org/10.54254/2755-2721/104/20240987

28. Zhang, B., Ma, R., Cao, Y., An, P. Swin-VEC: Video Swin Transformer-Based GAN for Video Error Concealment of VVC. The Visual Computer, 2024, 40(10), 7335-7347. https://doi.org/10.1007/s00371-024-03518-9

29. Zhang, K., He, Y., Yu, K., Liu, Y. Fast Data Processing Method for Multispectral Radiation Thermometry Based on Euclidean Distance Optimization. Optics Express, 2024, 32(2), 1342-1356. https://doi.org/10.1364/OE.510084

30. Zhang, Y., Xie, H., Zhuang, S., Zhan, X. Image Processing and Optimization Using Deep Learning-Based Generative Adversarial Networks (GANs). Journal of Artificial Intelligence and General Science, 2024, 5(1), 50-62. https://doi.org/10.60087/jaigs.v5i1.163

This article is an Open Access article distributed under the terms and conditions of the Creative Commons Attribution 4.0 (CC BY 4.0) License (http://creativecommons.org/licenses/by/4.0/).