SIN-CHN
SCIENTIFIC PRESS PTE.LTD.

Molecular & Cellular Biomechanics 2025, 22(4), 1206.
https://doi.org/10.62617/mcb1206

Article

Application of graphic design and animation technology based on biomechanics principle in communication optimization

Jiayi Xu[1,*], Chenchen Shan[1], Dan Huang[2]

[1] HBU-UCLan School of Media, Communication and Creative Industries of Hebei University, Baoding 130600, China
[2] School of Journalism and Communication of Hebei University, Baoding 130600, China
*** Corresponding author:** Jiayi Xu, 13362980348@163.com

CITATION

Xu J, Shan C, Huang D. Application of graphic design and animation technology based on biomechanics principle in communication optimization. Molecular & Cellular Biomechanics. 2025; 22(4): 1206. https://doi.org/10.62617/mcb1206

ARTICLE INFO

Received: 20 December 2024
Accepted: 14 January 2025
Available online: 18 March 2025

Abstract: With the development of information technology and digital media, graphic design and animation technology play an increasingly important role in modern communication. Through the combination and optimization of visual elements, graphic design realizes the efficient communication of information and the fast focusing of audience's attention, while animation technology makes complex information visualized and vivid by virtue of its dynamic performance and narrative ability, which enhances the comprehensibility and infectiousness of the content. This paper starts from the point of view of biomechanics discusses the application of graphic design and animation technology in information 7dissemination, analyzes the key technologies such as skeletal animation, motion capture and GPU-accelerated rendering, and proves its advantages in the fields of advertisement, education, and cultural dissemination through the comparison of specific application examples. The results of the study show that graphic design and animation combined with modern technology can significantly improve the effect of information dissemination and provide technical support and development direction for future communication innovation.

Keywords: graphic design; animation technology; information dissemination; skeletal animation; GPU-accelerated rendering; dissemination optimization

1. Introduction

In the modern society where information is highly developed, visual communication has become one of the important means of information communication [1]. With the rapid development of digital technology, the application of graphic design and animation technology in the field of communication has gradually penetrated into various aspects such as advertising, education, culture and entertainment [2]. With its intuitive and efficient visual performance, graphic design can quickly convey information and attract the attention of the audience, while animation, through dynamic narrative and vivid visual effects, makes complex information easy to understand and interesting, and further enhances the infectious force and memorability of the communication content [3]. Especially in the current context of continuous innovation of multimedia and interactive technology, combined with GPU acceleration, machine learning and virtual reality and other cutting-edge technologies, the expressive power and communication effect of graphic design and animation have been significantly improved, providing a new space for information communication and a new path of development and practice [4]. Therefore, exploring the application of graphic design and animation technology in communication optimization not only has theoretical value, but also provides important practical guidance for the innovation and optimization of information communication.

2. The role of graphic design and animation in communication optimization

2.1. Application of graphic design in communication

Graphic design is an important and indispensable tool in information dissemination, which simplifies and strengthens complex information through visual language, and delivers it to the audience in an efficient and intuitive way [5]. Compared with traditional text expression, graphic design can attract the attention of the audience in a very short time, enhance the comprehensibility and memorability of the information, and greatly improve the communication effect [6]. Whether in digital media and advertising, using simple and powerful graphic symbols to strengthen brand image and marketing information, or in the field of education and science popularization, through data visualization and illustration to present abstract knowledge intuitively, helping the audience to understand the content faster, graphic design plays a crucial role [7]. In addition, in cultural and artistic communication, graphic design combines traditional cultural elements with modern design concepts to reshape the expression of cultural symbols and expand the depth and breadth of cultural communication. With the rapid development of digital technology, graphic design has also shown great potential in emerging forms of communication such as dynamic graphics, Augmented Reality (AR) and Virtual Reality (VR), and has become the core means of information communication and interactive experience [8].

2.2. Application of animation in communication

As a form of dynamic visual expression, animation has unique advantages in information dissemination. It visualizes abstract and complex information through coherent images and vivid narratives, enhances the interest and comprehensibility of the content, and then enhances the attention and emotional resonance of the audience [9]. In advertising communication, animation can show product characteristics through creative visual effects and smooth movements, strengthen brand memory and attract consumers [10]; In the field of education and popularization of science, animation transforms boring knowledge points into vivid pictures, and helps audiences to quickly understand and master the content through dynamic demonstration process and interactive design [11]. In cultural communication, animation makes cultural content more acceptable and realizes wide dissemination through modern interpretation of traditional stories, symbols and elements [12]. In addition, animation shows stronger expressiveness and immersion in emerging digital media, Virtual Reality (VR) and Augmented Reality (AR) technologies, providing users with a more interactive and three-dimensional communication experience. With the development of technology, animation not only occupies an important position in the entertainment industry such as film and television, games, etc., but also plays a more and more crucial role in commercial publicity, knowledge dissemination and cultural inheritance [13].

3. Key technologies

3.1. Figure neural network

Graph Neural Networks (GNNs) are a type of deep learning model specifically designed to process graph-structured data. Graph data includes nodes and edges, such as social networks, molecular structures, recommendation systems, etc. Attention mechanisms play a key role in graph neural networks, helping models better focus on and utilize important information in graphs [14].

The Squeest-and-Excitation Network (SENet) is a representative example of the application of attentional mechanisms to channel dimensions in computer vision. It weights the feature graph by learning the weights of different channels to pay attention to the more important channel features and ignore the less important channel features [15]. Its structure is shown in **Figure 1** below.

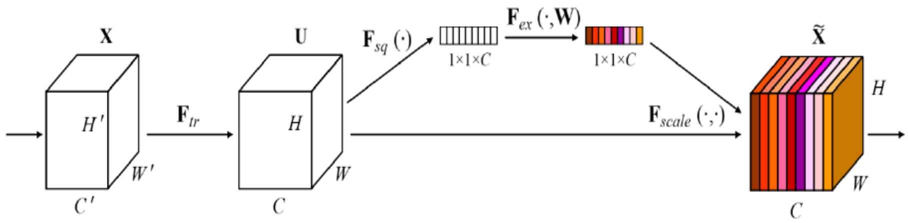

Figure 1. Structure of an SE block.

As shown in **Figure 1**, the SE-Net attention mechanism can be divided into three steps in the order of execution: Compression, activation, and reloading.

New empowerment. Each of the three steps is described below:

1) Compression: The characteristic input diagram of $C \times H \times W$ is compressed into $C \times 1 \times 1$ by using global average pooling to obtain the global receptive field. The process can be described by Equation (1).

$$Z_c = F_{sq}(u_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i,j) \quad (1)$$

where H and W represent the height and width of the feature map, and c represents the value with coordinate i, j in channel $u_c(i,j)$.

2) Activation: In order to learn the correlation of different channel feature graphs, a two-layer MLP layer is used here for nonlinear transformation.

$$s = F_{ex}(z, W) = \sigma\big(g(z, W)\big) = \sigma\big(W_2 \delta(W_1 z)\big) \quad (2)$$

where, and represent activation function in concrete implementation are RuLU and Sigmoid respectively, $W_1 \in R^{\frac{C}{r} \times C}$ and $W_2 \in R^{C \times \frac{C}{r}}$ are weight matrix, where r is scaling parameter, mainly used to reduce network computation.

3) Re-weighting: After the activation of the network, the weights of each channel of the feature map are obtained. The Sigmoid activation function is used to normalize the extracted weights and multiply them with the input features to

strengthen the important feature map and weaken the non-important feature map. The specific process can be described by Equation (3).

$$\tilde{x}_c = F_{scale}(u_c, s_c) = s_c \times u_c \tag{3}$$

where, s_c is the activation value of channel c in s, and u_c is the feature map of channel c in input feature map u, representing pixel-by-pixel multiplication. The whole operation can be regarded as using the learned weight coefficient of each channel to weight each channel of the input feature graph.

3.2. Virtual reality and augmented reality

Virtual Reality (VR) and Augmented Reality (AR) are important breakthroughs in the field of information dissemination and interaction technology in recent years. VR immerses users in the digital world by building a completely virtual environment; AR, on the other hand, enhances the expressiveness of the real world by overlaying digital information on real scenes. These two technologies can effectively improve the efficiency of information dissemination and user experience [16].

3.2.1. VR system based on graphics rendering

VR technology mainly relies on real-time graphics rendering technology to generate high-fidelity virtual scenes through geometric modeling, light calculation and texture mapping [17]. In the VR system, the Phong illumination model is commonly used for local illumination calculation, and the formula is as follows:

$$I = I_a + I_d(\mathrm{N} \times \mathrm{L}) + I_s(\mathrm{R} \times \mathrm{V})^n \tag{4}$$

Among them:

I : Final light intensity;

I_a: Ambient light intensity;

I_d: The intensity of diffuse reflection depends on the normal vector N and the direction of the light source L;

I_s: The intensity of specular reflected light depends on the reflection direction R and the observation direction V;

n : Sharpness of specular highlights.

3.2.2. Spatial localization and tracking techniques for AR

The core of augmented reality is the real-time fusion of virtual information and the real world [18], and key technologies include:

SLAM: Enables real-time 3D environment reconstruction and camera positioning.

Perspective projection model: Superposition of virtual objects with real environment is usually based on perspective projection matrix

$$P = K[R \mid t] \tag{5}$$

Among them:

P: Projection matrix;

K: The camera internal reference matrix;

R: A rotation matrix that describes the rotation of the camera coordinate system with respect to the world coordinate system;

t: Translation vector.

3.2.3. Gesture recognition and interaction

Based on computer vision and deep learning algorithms, VR/AR systems are able to recognize user gestures and realize natural interaction [19]. Commonly used gesture detection methods include:

Convolutional Neural Network (CNN): Extracting hand features.

Skeletal joint point tracking: Calculates motion vectors from joint point coordinates:

$$\Delta \mathrm{P}_i = \mathrm{P}_i^{t+1} - \mathrm{P}_i^{t} \tag{6}$$

Among them:

P_i^t: Coordinates of the key point of the gesture at time t;

$\Delta \mathrm{P}_i$: Motion vectors for key points.

3.2.4. Visual feedback and immersion

By optimizing the frame rate (FPS) and latency (Latency), the VR/AR system improves visual smoothness and reduces user vertigo. The optimization goal Equation is:

$$T_{\text{latency}} < 20 \text{ ms} \tag{7}$$

where T_{latency} is the latency of the system, which is usually required to be less than 20 ms to ensure user immersion.

3.3. Skeletal animation

3.3.1. Principles of skeletal animation technology

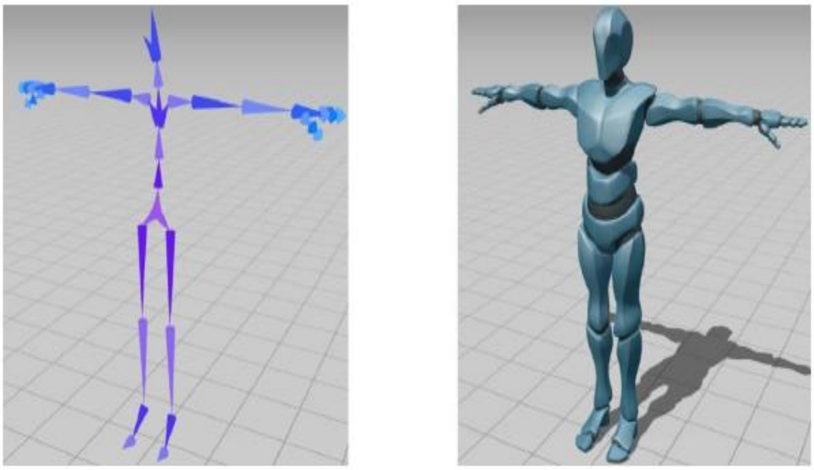

Figure 2. Skeleton (left) and skin (right).

In skeletal animation, an object usually consists of multiple skeletal nodes, which can form a skeletal chain or a skeletal hierarchy connected by multiple skeletal chains. Each bone node has its own coordinate system, and its position and rotation relative to the parent bone, and the corresponding object motion can be obtained by transforming these nodes [20]. As shown in **Figure 2**, the figure demonstrates the T-pose attitude of the human skeleton. Similar to the real-world skeleton, the motion of each bone in the skeleton changes the shape and position of the skin to which it is attached. Therefore, when animating a bone, the corresponding skin will also be animated to represent the current pose of the bone.

Like the basic model, the skeleton has its own corresponding transformation space, called the skeleton transformation space. Similarly, bones also have a local transformation space. As shown in **Figure 3**, the skeleton transformation is essentially a series of matrix space transformations.

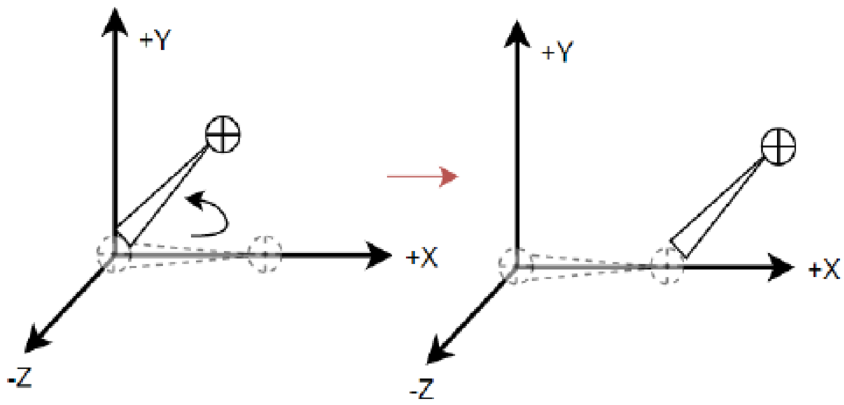

Figure 3. Bone rotation (left) and bone translation (right).

For the bone chain, determining a pose requires calculating the combined transformation of multiple bones. In the case of the arm, when the bones in the upper arm are rotated, the bones in the lower arm and hand also change. Similarly, when the lower arm is rotated, the bones of the hand will rotate with it, but the upper arm will not rotate. Therefore, to determine the posture of the role, it is necessary to first determine the result of each target bone transformation, and then determine the local transformation of all the parent bone nodes through the combination of bone transformations to determine the final position. The transformation process with the combination of arm bones is shown in **Figure 4**:

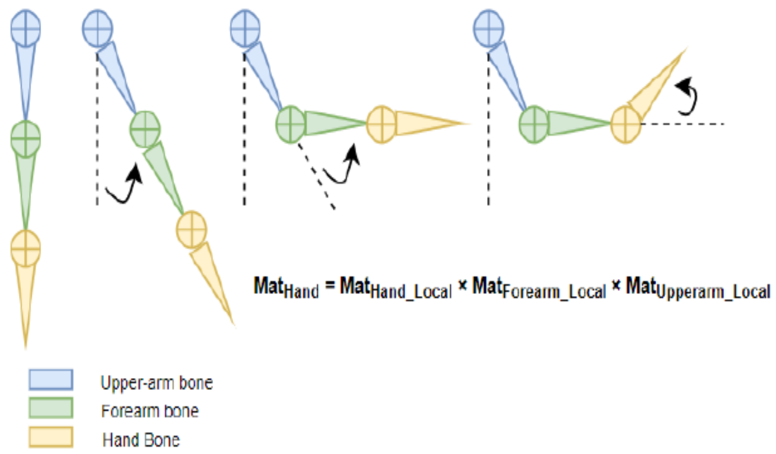

Figure 4. Rotation transformation process of arm bone.

After determining the posture of the character's skeleton, how to make the skeleton move is the key problem in the skeleton animation. In bone animation, a transformation state at a certain point in time is called a key frame. Key frame technology is a commonly used animation production method. By recording the pose and position information of the object on the key frame, and calculating the intermediate frame animation of the object through interpolation algorithm between each key frame, continuous animation effect can be achieved. The position and rotation information of the keyframe is usually obtained by the animator through hand-crafting or recording the action. As shown in **Figure 5**, taking the arm skeleton as an example, with the time interval of 2.5 s as the key frame, smooth transition between animation frames can be completed by interpolation:

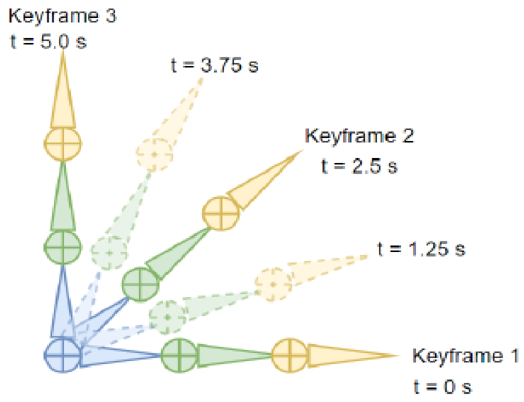

Figure 5. Keyframe interpolation.

3.3.2. Analysis of the relationship between skeletal node movement and biomechanics

In skeletal animation production, realism and naturalness are one of the core indicators for evaluating animation quality. In order to realize high-quality animation effects, it is necessary to study the motion law of skeletal nodes in depth and integrate the key factors in the real biomechanical structure into the animation production. The following will be analyzed in detail from the aspects of muscle force points, joint range of motion, and mechanical principles.

1) Relationship between muscle force points and bone movement

In human or animal models, muscle contraction and extension is the main power source to drive bone movement. The muscle force point determines the direction and amplitude of bone movement. For example, when modeling the movement of the human arm, the forces of the biceps and triceps muscles influence the rotation and movement of the forearm through the elbow joint. The muscle mechanics action can be expressed as the following Equation (8):

$$\tau = F \times d \times \sin(\theta) \tag{8}$$

where: τ denotes the torque of the joint; F denotes the magnitude of the force on the muscle; d denotes the distance from the point of force on the muscle to the center of rotation of the joint; and θ denotes the angle between the direction of force on the muscle and the bone.

2) Mechanical constraints on the limits of movement

The limit of skeletal movement depends not only on the strength of muscles, but is also limited by the structure of joints. For example, the range of motion of the human knee is mainly focused on flexion (0°–135°) and extension (135°~0°), while the range of abduction and internal rotation is very limited. This limitation is due to the bone morphology of the joint, the stretching range of the ligaments, and the location of the tendon attachment points. Skeletal animation should be designed with the following factors in mind:

Range of Motion (ROM): Take the knee joint as an example, its flexion range of motion is about 0°–135°.

Joint torque: For example, the ankle joint needs to generate a large torque to balance the body when walking.

Joint stability: Some joints (e.g. shoulder joint) have a large range of motion, but their stability depends on the surrounding muscle groups and ligament structures.

3) Rotation limits and constraints for skeletal nodes

Skeletal nodes in animation usually need to add rotational constraints to conform to the characteristics of real biological movements. Take the shoulder joint as an example, its range of motion is as follows:

Flexion: 0°–180°;

Extension: 0°–60°;

Abduction: 0°–180°;

Internal rotation: 0°–70°.

By adding rotational constraints to the skeletal animation, we can avoid uncoordinated movements caused by over-rotation of the joints. In addition, mechanical constraints need to be imposed on the rotation in order to ensure the physical rationality of the movement. For example, when the arm is lifted to 180°, due to the mechanical properties of the shoulder muscles, its force point and moment change, and the trend of gradually decreasing force needs to be reflected in the animation.

4) Optimize the naturalness of combining muscle and bone dynamics

In order to further improve the naturalness of the skeletal animation, the muscle dynamics can be simulated to influence the bone movement, the specific implementation steps are as follows:

First, an elastic model based on stretching and contraction is used to accurately simulate the dynamic changes of muscles in the process of movement during the simulation stage of muscle dynamics, such as in the running animation, the dynamic alternation of quadriceps and hamstrings plays a key role in the leg bone movement.

Secondly, a mechanical feedback mechanism is established to link the dynamic change of muscle force with the motion of skeletal nodes, and the moment calculation of joints is used to drive the skeletal motion, so that the skeletal motion is more in line with the real biomechanical characteristics.

Finally, based on the gait analysis data in biomechanics research, such as step length, step frequency and center of gravity trajectory, gait optimization is carried out on the rhythm of the character's running animation to ensure the smoothness and naturalness of the animation. Through the above steps, the accurate simulation of muscle dynamics on bone movement is realized, so as to enhance the overall naturalness of the skeletal animation.

5) Application of biomechanical research data in skeletal animation

In order to ensure the scientificity and authenticity of skeletal animation, you can refer to the biomechanical research data in **Table 1** below:

Table 1. Scientific and biomechanical data table of skeletal animation.

joint	Range of motion/torque
Knee joint	Buckle 0°–135°, internal/external rotation ± 15°
hip joint	Buckle 0°–120°, internal rotation ± 45°
elbow joint	Buckle 0°–150°, rotate ± 90°
shoulder joint	The average torque during abduction is 30 Nm
Knee joint	The average torque during running is 60 Nm

The relationship between the speed of muscle contraction and force can be expressed as

$$F = \frac{F_{max}(v_{max} - v)}{v_{max} + k \times v} \tag{9}$$

where F_{max} is the maximum muscle contraction force, v_{max} is the maximum contraction velocity, v is the current contraction velocity, and k is a parameter for the velocity-force relationship.

In character design, these can be used to limit the range of motion of skeletal nodes based on the joint range of motion and dynamically optimize the realism of character movements through muscle forces.

4. Implementation method of communication optimization technology

4.1. Graphic design optimization technology

4.1.1. Vector graphics and rendering optimization

Vector graphics is a graphical representation of Points, lines, and curves described by mathematical formulas, and its core relies on the following geometric elements: Firstly, Control Points are the key defining points of a curve, such as in Bezier curves, which determine the shape and direction of the curve; Secondly, the path and edge are formed by the connection of control points, which can be straight lines or curve segments to build the overall structure of the graph. Finally, properties such as color, transparency, lighting characteristics, etc. further enrich the visual representation of vector graphics. Vector graphics have significant advantages over pixel-based bitmap graphics, the most critical of which is its lossless scaling properties: Because vector graphics rely on mathematical descriptions rather than fixed pixel data, there is no distortion when zooming in or out, and at the same time, the amount of data in vector graphics is relatively small, which makes it have low bandwidth and storage costs during storage and transmission. However, vector graphics also face some challenges, especially when dealing with high-precision curves and complex scenes, the overhead of real-time rendering is large; In addition, the calculation of light and shadow relies on point-by-point characteristics, resulting in slow rendering speed; In the process of dynamic vector graphics and animation, efficient data reconstruction and feature extraction algorithms are needed to cope with the frequently changing graphic requirements. To address these challenges, this paper proposes an innovative vector graphics rendering optimization method that combines attention mechanisms with graph neural networks (GNN). By introducing an attention mechanism, the model can focus more effectively on key features in the graphics, thus optimizing the allocation of computational resources during rendering. The graph neural network uses its powerful graph structure processing ability to realize the modeling and optimization of complex geometric relations in vector graphics, so as to ensure the rendering quality and significantly improve the rendering efficiency.

(1) Graphical data representation and geometric modeling

Representation of vector graphics as weighted graph data structures $G = (V, E, X)$:

V : A collection of nodes representing control point coordinates;

E : A collection of edges that represents the connection of lines or curves;

X : Additional features of nodes and edges, including colors, normal vectors, light attributes, etc.

Bessel curve modeling:

Common quadratic and cubic Bessel curve equations can be represented by the following Equations (10) and (11):

Quadratic Bessel curve:

$$B(t) = (1-t)^2 P_0 + 2t(1-t)P_1 + t^2 P_2, \quad t \in [0,1] \tag{10}$$

Cubic Bessel curves:

$$B(t) = (1-t)^3 P_0 + 3t(1-t)^2 P_1 + 3t^2(1-t)P_2 + t^3 P_3, \quad t \in [0,1] \tag{11}$$

where P_0, P_1, P_2, P_3 is a control point and t is a curve interpolation parameter in the range [0,1] for sampling discrete points on the curve.

The curve is decomposed into a discrete set of points $V = \{B(t_i)\}$, through which rendering and lighting calculations are performed to reduce the computational complexity of the curve.

(2) Attention mechanism to optimize feature extraction

In order to reduce the computational overhead of unimportant features, the SE attention mechanism is introduced to weight and optimize the graphical data channel features.

Global feature compression:

The input feature graph $X \in \mathbb{R}^{C \times H \times W}$ is extracted from the global description by global average pooling:

$$z_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} X_c(i,j), \quad c = 1,2,\ldots,C \tag{12}$$

z_c: Global average of channel c.

H and W: The height and width of the sign map.

$X_c(i, j)$: The value of pixel (i, j) in channel c of the feature map.

Nonlinear activation and weight learning:

Computation of channel weights via two-layer fully connected networks W_1 and W_2:

$$s = \sigma\big(W_2\sigma(W_1 z)\big) \tag{13}$$

where σ is the ReLU activation function and σ is the Sigmoid function.

Channel Re-weighting:

Weights each channel of the input feature map X :

$$X'_c = s_c \times X_c \tag{14}$$

X_c: Channel c of the original feature map.

$X_c^{'}$: Re-weighted feature maps where important features are amplified and unimportant features are weakened.

By weighting the feature passes through the attention mechanism, graphical features that are critical to the rendering process can be highlighted and redundant computations can be reduced.

(3) GNN accelerates light and shadow rendering

Lighting computation (e.g., Phong model) is often a bottleneck in vector graphics rendering. By introducing Graph Neural Network (GNN), the lighting features are efficiently propagated and predicted among nodes to reduce the redundancy of point-by-point computation.

Node feature initialization:

Initialize node features as coordinates and normal vectors:

$$h_i^{(0)} = [\mathrm{p}_i, \mathrm{n}_i] \tag{15}$$

p_i: Node coordinates.

n_i: Node normal vector for light direction calculation.

Message passing and feature update:

In GNN, node light features are updated through message passing mechanism:

$$h_i^{(l+1)} = \sigma(Wh_i^{(l)} + \sum\nolimits_{j\in\mathcal{N}(i)} \alpha_{ij} W_e h_j^{(l)} \tag{16}$$

Among them:

W, W_e: Eigen-transformation matrix of nodes and edges.

$\mathcal{N}(i)$: The set of neighboring nodes of node i.

α_{ij}: The attentional weights of the edges, calculated by Equation (17):

$$\alpha_{ij} = \frac{\exp\Big(\mathrm{LeakyReLU}\big(a^T\big[Wh_i||Wh_j\big]\big)\Big)}{\sum_{k\in\mathcal{N}(i)} \exp\,(\mathrm{LeakyReLU}(a^T[Wh_i||Wh_k]))} \tag{17}$$

Lighting feature output:

After propagation through several layers of GNN, the node illumination feature $h_i^{(L)}$ is directly used for light rendering:

$$L_i = I_a + I_d(\mathrm{n}_i \cdot \mathrm{n}_i) + I_s(\mathrm{r}_i \cdot \mathrm{v})^n \tag{18}$$

With the node feature propagation of GNN, the redundancy of lighting computation can be drastically reduced and the rendering efficiency can be improved.

4.1.2. Machine learning-aided design

Machine learning, as a powerful data-driven approach, is revolutionizing every aspect of graphic design. Through the learning of a large number of training data and models, machine learning can realize intelligent generation and prediction, for example, using Generative Adversarial Network (GAN) and other generative models to automatically create design elements, such as ICONS, textures, etc., thus greatly improving the creativity and efficiency of design. At the same time, machine learning also shows great potential in design optimization, through regression and classification models to evaluate the effect of design schemes, designers can quickly iterate and optimize design schemes to achieve more accurate design decisions. In addition, machine learning also promotes the development of intelligent interaction through natural language processing and computer vision technology, so that design tools can understand and respond to the needs of designers, and achieve more intelligent and humane operations, thus significantly improving the efficiency of design work and user experience.

(1) Generative Adversarial Network (GAN) is used for automatic graphics generation

Generative Adversarial Network (GAN) is an important method in current machine learning generation tasks. It is composed of two adversarial networks: generator and discriminator:

Generator G: Receive random noise z and output the generated image data:

$$x_{\text{gen}} = G(z; \theta_g) \tag{19}$$

Discriminator D: Determine whether the input data is real or generated and output the probability value $D(x)$:

$$D(x) \in [0,1] \tag{20}$$

where x is the input data. The discriminator network parameter is θ_d.

Objective function: The optimization objective of GAN is to make the image generated by generator G as close as possible to the real data, so that discriminator D cannot distinguish between the two. The objective function is:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{\text{data}}}[\log D(x)] + \mathbb{E}_{z \sim p_z}\left[\log\left(1 - D\left(G(z)\right)\right)\right] \tag{21}$$

Among them:

p_{data}: Real data distribution;

p_z: Input noise distribution.

The main function is to generate vector graphics with consistent style based on input conditions. Generate high quality texture and lighting effects.

(2) Deep neural networks for graphic design optimization

In vector graphic design, quality evaluation and optimization of design results can be achieved by regression model or classification model. Let the features of design scheme X be the input and design quality Y be the output, and predict the design effect by regression model:

$$Y_{\text{pred}} = f(X; \theta) \tag{22}$$

where: Y_{pred} is the predicted design quality score. f is the neural network model with parameter θ.

Loss function: minimizes the mean square error between the true value Y and the predicted value Y_{pred}:

$$L(\theta) = \frac{1}{N}\sum_{i=1}^{N}\left(Y_i - f(X_i; \theta)\right)^2 \tag{23}$$

Classification of design styles can be done by Convolutional Neural Networks (CNN) to recognize the style category of the input graphic:

$$LP(y|X) = \text{softmax}(W \cdot \text{CNN}(X) + b) \tag{24}$$

Among them:

$P(y|X)$: Enter the probability that graph X belongs to category y.

W and *b*: weights and biases of the fully connected layer.

$\text{CNN}(X)$: Feature extractor for extracting high dimensional features from input graphics.

The main function is to identify design styles (e.g., flat, anthropomorphic). Adjust the graphic design scheme according to the prediction results to improve the design quality.

(3) Reinforcement Learning assisted intelligent design tool

In the design interaction process, Reinforcement Learning (RL) can be used to intelligently optimize the design process. The core components of Reinforcement Learning include:

State of affairs S : Current design program features.

Movements A : Operation of design tools, such as adjusting colors, adding elements, etc.

Incentives R : Changes in quality scores after design program improvements.

Goal: Automatic design optimization by maximizing the cumulative reward function R :

$$\max_{\pi}\mathbb{E}\left[\sum_{t=0}^{T}\gamma^t R_t\right] \tag{25}$$

Among them:

π : The strategy function that determines the action A_t to be taken in state S_t .

γ: Discount factor for balancing short-term and long-term rewards.

R_t: Instant rewards earned at time step *t*.

The main function is to intelligently adjust the position of elements to optimize the design layout. Real-time feedback through user interaction improves design efficiency.

Machine learning provides intelligent solutions for automatic generation, quality assessment and optimization of graphic design through Generative Adversarial Networks (GAN), Deep Neural Networks (DNN) and Reinforcement Learning (RL).

These techniques can dramatically improve design efficiency and meet real-time propagation requirements.

4.2. Animation optimization technology

4.2.1. Bone animation and motion capture

After binding the graph data to the skeleton, the three-dimensional motion characteristics of the skeleton must be transferred to the skeleton in order to realize the process of driving the modeling action through the skeleton animation. This process is mainly realized through node mapping and projection. First, the graph node is mapped to the bone, and the mapped point after the movement is obtained based on the movement trend of the bone. Then, the three-dimensional coordinate after the movement is calculated according to the mapping relationship, and finally the action sequence of the modeling is obtained by projecting it onto the XY two-dimensional plane.

(1) Node mapping

After the binding between the skeleton and the graph is completed, in order to transform the graph with the skeleton, and to ensure that the shape of the graph after transformation remains unchanged, the relationship between the graph node and the bone is first determined, that is, the graph node data of the composition of the shape is mapped to the bone. Since the Z coordinate of the skeleton and shape figure is 0 in the initial state, the node mapping process of the first step is unfolded in the XY plane. As shown in **Figure 6**.

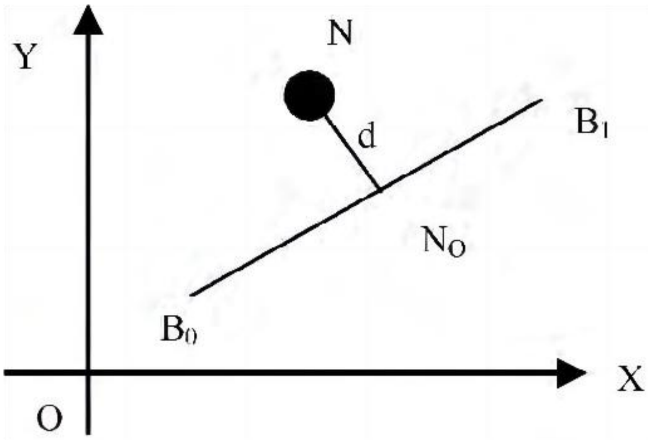

Figure 6. Schematic diagram of bone rotation around the Y-axis.

In this case, the relationship between the point and the line is determined by the distance d between them. If you think of the skeleton as B_0 straight line, B_0 and B_1 are known, assume that the equation of the line in the XY plane is:

$$ax + by + c = 0 \tag{26}$$

Then set the coordinate of point *N* as $(X_N, Y_N, Z_N), Z_N = 0$, then according to the distance formula from point to line, we can get:

$$d = \frac{| ax_N + by_N + c |}{\sqrt{a^2 + b^2}} \tag{27}$$

Knowing the distance from the point to the bone in the plane and the equation of the line, one can find the coordinates of the point N_0, the mapping point of the point N on the bone B, set as $(X_{NO}, Y_{NO}, Z_{NO}), Z_{NO} = 0$.

(2) Transformation of mapping point

After completing the mapping of modeling nodes to bones, the coordinates of the mapping points can be obtained after the change according to the change of the skeleton's action. The movements of the bones are extracted from the database. In this paper, we only take the rotation transformation as an example to illustrate this process.

Suppose the bone is centered at its vertex B_0 and rotated around the Y-axis by an angle β, as shown in **Figure 7**.

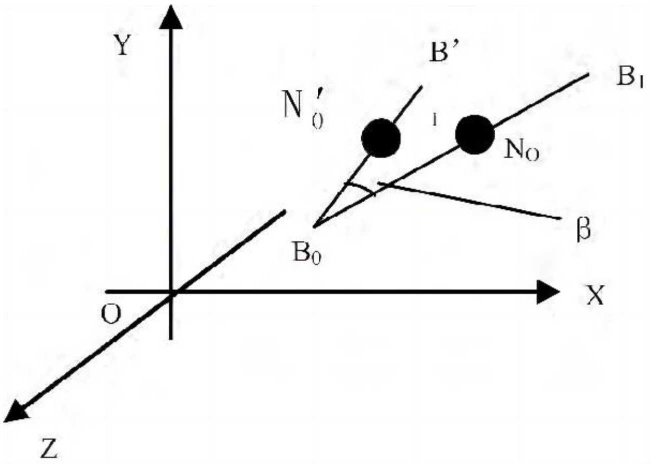

Figure 7. Schematic diagram of bone rotation around the Y-axis.

Then the coordinates of $N_0^{'}$ can be calculated by the following formula

$$\begin{bmatrix} X'_{No} \\ Y'_{No} \\ Z'_{No} \end{bmatrix} = \begin{bmatrix} X_N^0 \\ Y_N^0 \\ Z_N^0 \end{bmatrix} + \begin{bmatrix} d_{BN} \times \cos\beta \\ 0 \\ d_{BN} \times \sin\beta \end{bmatrix} \tag{28}$$

where d_{BN} represents the distance between B_0 and N_0. Similarly, if the bone rotates α、β、γ angles about the X, Y, and Z axes, then the $N_0^{'}$ point after N_0 rotation is expressed as:

$$\begin{bmatrix} X'_{No} \\ Y'_{No} \\ Z'_{No} \end{bmatrix} = R_\alpha \cdot R_\beta \cdot R_\gamma \cdot \begin{bmatrix} X_{NO} \\ Y_{NO} \\ Z_{NO} \end{bmatrix} \tag{29}$$

where $R_\alpha \cdot R_\beta \cdot R_\gamma$ is the rotation matrix.

(3) Skinning

The skinning technique binds mesh vertices to bones, which are animated by Linear Blend Skinning (LBS):

$$\mathrm{v}'_i = \sum_{j=1}^{n} w_{ij}\, M_j^{\mathrm{global}} \mathrm{v}_i \tag{30}$$

Among them:

v_i: The initial coordinates of the i th vertex of the mesh;

v_i': The coordinates of the vertex after the skeleton transformation;

w_{ij}: Binding weights of vertex v_i and joint j ($\sum w_{ij} = 1$);

M_j^{global}: Global transformation matrix of joint j.

Use Dual Quaternion Skinning (DQS) to address possible mesh volume collapse in LBS.

(4) Motion capture

Motion capture (MoCap) is a technique that records human or object motion data and applies it to digital character animation generation. Its main workflow includes: First, data acquisition through sensors, cameras, or marker points to record the details of the character's movements; then, pre-processing of the captured raw data, including filtering and de-noising and completing the missing data to ensure the accuracy and completeness of the data; then, mapping the processed motion data to the digital skeleton structure, a process known as skeleton mapping; and finally, driving mesh models through joint transformations to drive the mesh model, thus producing smooth animation effects. Motion capture techniques are divided into three main categories: Optical capture, inertial capture, and deep learning-assisted capture. Optical capture utilizes a camera to capture the motion trajectory of a marker point, which has the advantage of high precision and is suitable for complex movements, but has the disadvantage of being susceptible to occlusion, which may lead to data loss. Inertial capture uses inertial sensors (e.g., accelerometers, gyroscopes) to record motion data, which has the advantage of being unaffected by occlusion and good real-time performance, but suffers from the problem of cumulative error. Deep learning-assisted capture, on the other hand, utilizes deep learning models (e.g., gesture estimation networks such as OpenPose, DeepLabCut, etc.) to predict the key points of the human body from the video, which leads to more efficient and accurate motion capture.

The pose estimation model Equation (31): Let the input video frame I and the output key point coordinate set $P = \{p_i\}$ be:

$$P = \underset{p}{\mathrm{argmax}} \sum_{i=1}^{n} \log P(p_i | I; \theta) \tag{31}$$

where:

$P(p_i|I;\theta)$ is the probability distribution of keypoint p_i in input frame I. θ is the model parameters.

4.2.2. GPU accelerated rendering

(1) Rendering pipeline

GPU rendering is based on the graphics rendering pipeline, which is divided into the following main phases:

Vertex processing stage

Input: geometric data (vertex coordinates, normal vectors, etc.).

Task: Perform model transformations and lighting calculations to generate cropped space coordinates.

Vertex transformation formula:

$$v' = P \times V \times M \times v \quad (32)$$

Among them:

v: The coordinates of the input vertices;

M: Model transformation matrix;

V: View transformation matrix;

P: Projective transformation matrix.

Rasterization phase:

Converts vertex data to screen pixels (Fragment) and generates depth values.

Goal: Determine which pixels need to be drawn.

Fragment processing phase:

Executes the Fragment Shader, which performs shading and lighting calculations for each pixel.

Lighting calculation (Phong model):

$$I = I_a + I_d(n \times l) + I_s(r \times v)^n \quad (33)$$

(2) Shader optimization

Shader is a key program in the GPU rendering process, mainly including:

Vertex shader: Used for transforming vertex coordinates and normal vectors.

Slice shader: Used for lighting, texture mapping and other pixel-level calculations.

Optimization methods:

Reduce the number of texture accesses: Reduce access latency by caching texture data.

Parallel execution of lighting calculations: Take advantage of the multi-threaded nature of the GPU to distribute lighting calculation tasks.

Batch rendering (Instancing): Repeatedly render similar objects to reduce the communication overhead between CPU and GPU.

(3) GPU parallel computing model

GPU uses SIMD (Single Instruction, Multiple Data) architecture, suitable for parallel task decomposition, mainly in:

Thread grouping: The task is divided into thread blocks (Thread Block) and grids (Grid), and executed in parallel:

$$N = \text{Grid Size} \times \text{Block Size} \quad (34)$$

Chunked illumination computation: Distribute the illumination computation to multiple threads in parallel, such as the pixel-level computation of the Phong illumination model.

The schematic formula is shown below:

$$L_{\text{total}} = \sum_{i=1}^{n} T_i \times (I_a + I_d(n_i \times l_i) + I_s(r_i \times v_i)^n) \quad (35)$$

Among them:

T_i: Thread i ‘s assigned tasks;

L_{total}: Overall rendered lighting results.

(4) Optimization methods for GPU-accelerated rendering

View cone culling: Cull geometric data that is not in the view cone to reduce unnecessary rendering calculations.

Let the vertex coordinate be P. If the following conditions are satisfied, it is culled:

$$-1 \leq p_x/p_w \leq 1, -1 \leq p_y/p_w \leq 1, 0 \leq p_z/p_w \leq 1 \quad (36)$$

where p_w is the chi-square coordinate of the vertex after the projection transformation.

Delayed Rendering: Reduces redundancy in lighting computation by processing geometry data and lighting data in stages.

Phase 1: Geometry rendering, storing normal vector, depth, and material information into multiple buffers (G-buffers).

Stage 2: Lighting computation, which is performed only on the visible pixels on the screen.

GPU-accelerated rendering dramatically improves the efficiency and real-time performance of animation and graphic design through highly parallel computing power and optimized rendering pipelines. Through shader optimization, parallel lighting computation and culling techniques, GPU achieves high-quality rendering effects in complex scenes and provides strong technical support for propagation optimization.

4.3. Biomechanical data-driven animation rendering optimization

4.3.1. Application of animal movement velocity and acceleration laws

According to biomechanics research, different animals have specific velocity, acceleration and posture change patterns when moving. For example, there are significant differences in the time scaling and acceleration characteristics of limb movements of horses under different gaits (e.g., walking, jogging, and galloping). These patterns can be used to optimize keyframe settings and frame rate allocation in animation rendering:

Velocity and acceleration dynamic adjustment: In the animation, by extracting the velocity change curve and acceleration data of the biological motion, it is applied to the interframe transition of the animation. The equations are as follows:

$$v(t) = v_0 + at \quad (37)$$

$$s(t) = v_0 t + \frac{1}{2} a t^2 \quad (38)$$

where $v(t)$ is velocity at time, a is acceleration, and $s(t)$ is displacement. The inter-frame motion smoothness of the character at different gaits is adjusted by Equations (37) and (38).

Keyframe optimization: Adjust the keyframe density according to the importance of different movement stages. For example, when a horse is galloping, the keyframes for landing and lifting the limbs need to be denser, while the keyframes for body balance adjustment can be appropriately reduced. This not only optimizes the rendering efficiency, but also improves the smoothness of the motion.

GPU animation frame rate optimization: Dynamically adjust the frame rate in GPU-accelerated rendering by analyzing the changing rules of biological acceleration and posture in real time. For example, the frame rate can be increased to 60 FPS in the fast-changing motion phases (e.g., jumping and turning) and reduced to 30 FPS in the smoother phases (e.g., running at a constant speed), thus saving computational resources while maintaining the smoothness of the animation.

4.3.2. Modeling the change law of body posture

The body posture changes of animals in motion are affected by muscles, bones and inertia. By analyzing these laws, the pose modeling and rendering process in animation can be optimized:

Pose curve fitting: The pose changes of biological models are fitted using motion capture data and pose curves. For example, the arching and sinking of the back of a horse while running can be realized by cubic spline interpolation calculation:

$$\mathrm{P}(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$

where $\mathrm{P}(t)$ is the position of the back curve at time t and a_0, a_1, a_2, a_3 is the fitting parameter.

Inertia and balance control: By adding inertia parameters, the mechanism by which an organism maintains its balance during high-speed movements is simulated. For example, when running and steering, an animal usually tilts its body to counteract centrifugal force, and this tilt angle can be calculated by the following Equation (39):

$$\tan(\theta) = \frac{v^2}{r \times g} \tag{39}$$

where θ is the angle of inclination, v is the velocity, r is the radius of the turn, and g is the gravitational acceleration.

4.3.3. Simulation of animal skin elasticity

The skin of an organism undergoes elastic deformation when subjected to an external force, and the simulation of this deformation can make the organism in the animation more realistic. Skin elasticity can usually be modeled by an elastic deformation model:

$$F = -kx \tag{40}$$

where F is the elastic restoring force, k is the elastic coefficient and x is the shape variable. The specific realization method is as follows:

Dynamic texture mapping is created based on the thickness and elasticity coefficient of the skin in different parts of the body. For example, the skin of an animal's abdomen is looser, while the skin of its limbs is tighter. In GPU rendering, the elastic deformation calculation is decomposed into multiple threads to deal with the deformation of different parts separately to improve the rendering efficiency.

4.3.4. Simulation of bone toughness and mechanical properties

The toughness and hardness of bones determine the degree of deformation when they are subjected to external forces. In physical simulation, collision and deformation can be calculated by the material properties of bones with external forces:

Material property modeling: The bone is divided into two parts, rigid and semi-rigid, with the rigid part (e.g., long bone) having a high degree of stiffness and the semi-rigid part (e.g., cartilage) having a certain degree of flexibility.

Collision detection and response: Calculates the change in motion of the skeleton in case of collision based on the rigid-body dynamics formulation:

$$\vec{F} = m \times \vec{a}$$
$$\vec{\tau} = I \times \vec{\alpha} \quad (41)$$

where $\vec{F}$ is the external force, m is the mass, $\vec{a}$ is the acceleration, $\vec{\tau}$ is the moment, I is the moment of inertia, and $\vec{\alpha}$ is the angular acceleration.

5. Application examples

5.1. Virtual Reality (VR) and Augmented Reality (AR) applications

In Virtual Reality (VR) and Augmented Reality (AR) interactive scenarios, the immersion and interactivity of the user experience is crucial. In VR environments, users interact with virtual characters in depth, and this interaction requires a high degree of realism and immediacy; whereas in AR applications, by superimposing virtual elements (e.g., animated characters) on real environments, users are able to experience augmented visual effects in the familiar real world. To fulfill these demands, real-time rendering, low latency and smooth animation performance become key technical challenges. In this paper, we propose an innovative approach to significantly improve the performance of interactive scenes through GPU acceleration and skeletal animation optimization. The specific results are shown in **Table 2**.

Table 2. Evaluation indexes of virtual reality (VR) and augmented reality (AR) interactive scene effect.

index	Traditional method	Textual method	Analysis
Frame rate (FPS)	45	60	The method in this paper improves the frame rate and reduces lagging.
Render time (ms)	30	18	GPU accelerated rendering significantly reduces rendering time.
Animation smoothness score	65	92	Animation transitions are more visual and joint movements are smoother.
User immersion	lower	high	Smooth animation enhances the VR/AR immersion experience.

This method achieved a frame rate of 60 FPS and reduced rendering time to 18 milliseconds, which greatly improved the smoothness and real-time performance of the animation. This optimization not only enhances user immersion, but also ensures the smoothness and naturalness of the interactive experience, enabling users to enjoy a seamless interactive experience in an environment where virtual and reality converge. Through these technical improvements, the method presented in this paper provides strong support for interactive scenarios in VR and AR applications, driving the further development of immersive experiences.

5.2. Film and game animation production

In the field of film and game production, animators strive to create high-quality character animation, such as realistic character movements for film special effects and smooth character performances for games, which need to meet extremely high production standards. First of all, high-precision animation is one of the core requirements, and every action detail needs to be accurately presented to ensure that the character's image is lifelike. Secondly, complex lighting and shadow calculation is also a key link, accurate light and shadow effects can enhance the sense of reality and visual impact of the scene, so that the audience or players have more sense of inclusion. However, these high-quality production requirements are often accompanied by long render times, which can reduce production efficiency and increase costs. The method proposed in this paper effectively solves this problem. By using GPU parallel computing technology, it gives full play to its powerful data processing ability, and combines optimization algorithms to deeply optimize the entire animation production process. See **Table 3** for details.

Table 3. Evaluation indexes of film, television and game animation production effects.

index	Traditional method	Textual method	analyze
Render time (frames per second)	30 fps	60 fps	Gpu-accelerated rendering greatly improves animation rendering efficiency.
Light and shadow calculation time	Longer (delay)	Very short (real time)	Lighting computation is parallelized to reduce latency.
Smoothness and precision	moderation	Extremely high	GPU parallel computing makes character actions more refined and natural.
Resource occupancy rate	high	low	This method makes more efficient use of hardware resources.

With the help of this technology, the animation smoothness score is significantly increased to 92 points, which means that the animation's fluency and fidelity have reached a high level; At the same time, the rendering efficiency has also been greatly improved to 60 FPS, which can meet the demand for efficient production of high-quality animation in film and game production, and provide strong technical support for animators to help create more wonderful film and television works and game experience.

5.3. Optimize the graphic design and animation application scene with biomechanics

In the teaching of biomechanics courses, many knowledge points involve complex mechanical analysis and human movement laws. These contents are relatively abstract for students, especially when it comes to the stress distribution in skeletal movement, the mechanical mechanism of muscle contraction and energy conversion, etc., it is difficult to form an intuitive cognition by simply relying on traditional illustrations or explanations. Therefore, by combining graphic design and animation technology, these complex concepts can be visualized to help students understand biomechanical principles faster and more accurately.

5.3.1. Leverage principles in skeletal movements

Many movements in the human skeletal system can be viewed as applications of the lever system. For example, when the biceps flexes the arm, the elbow joint acts as a fulcrum, and the moment generated by the muscle force pushes the bones into motion. To help students understand the application of lever principles in biomechanics, an instructional animation can be designed in the following steps:

Animation display content: Demonstrate the elbow bending process through step-by-step animation, labeling the pivot point (elbow joint), power (pull force of muscle contraction) and resistance (weight of the forearm) respectively.

Dynamic mechanics analysis: Add real-time moment calculation formula to dynamically display the moment changes under different angles.

Comparison of traditional lever model: By comparing the skeleton with the lever model, students can intuitively understand the similarity between skeletal motion and lever mechanics.

5.3.2. Mechanics of muscle contraction

The mechanical changes of muscles in different motion states are very important for understanding human motion. Example:

Static animation: Labeling the contraction and stretching state of muscles through graphic design, marking the starting point and end point (tendon attachment point).

Dynamic animation: Show the change of tension on the bone when the muscle contracts, combined with a color gradient to indicate the amount of stress.

Mechanical process visualization: Embed simple biomechanical equations in the animation to show the relationship between muscle contraction speed and force in real time.

5.3.3. Joint range of motion and energy conversion

The animation can simulate the energy consumption and conversion of the human body in different movement states. For example:

Animation display: Combine with motion capture technology to produce animation of knee flexion and extension in the gait cycle, and demonstrate the distribution of energy consumption through graphical data.

Optimized content: Superimpose heat maps in the graphic design to visually display the joint pressure concentration area and the location of peak energy consumption.

5.3.4. Comparative analysis of optimized teaching animation and traditional teaching

Table 4. Comparative analysis of teaching.

Comparative indicators	Traditional teaching methods	Optimizing animation and graphic design methods
Student understanding efficiency	Relying on written and verbal descriptions, slow comprehension speed	Dynamic display of mechanical concepts, intuitive expression, faster understanding speed
Presentation of complex knowledge points	Repeated explanations are needed, which can easily cause confusion among students	Using animation to simulate the movement process of bones and muscles, reducing the difficulty of understanding
interactive quality	Low student participation, learning only through observation and memory	Animation combined with interactive design; students can manipulate models to test motion under different conditions
Teaching effectiveness evaluation	The test results fluctuate greatly, and the learning effectiveness of students is unstable	Data shows that students' memory and mastery rate of knowledge points have significantly improved
Student feedback	Abstract knowledge and dull learning	The content is vivid and interesting, and students provide feedback that the learning process is easier and more efficient

In order to evaluate the effect of optimization design in teaching, experiments can be designed to compare the advantages and disadvantages of traditional teaching with optimization design teaching. The following **Table 4** demonstrates the comparison of the two teaching methods:

Through the optimization of graphic design and animation technology, teaching can obtain a variety of significant advantages: First of all, to improve the quality of teaching, the combination of animation technology and graphic design can present complex biomechanical concepts in a more intuitive form, to help students better grasp the knowledge points from the dynamic changes. Secondly, enhance students' interest in learning, compared with the traditional boring teaching methods, vivid image animation content can effectively stimulate students' interest in learning and enhance the sense of classroom participation. In addition, to improve the mastery rate of knowledge points, the dynamic demonstration in the animation combined with real-time data analysis, so that students can more deeply understand the key principles of the movement process. Finally, it supports personalized learning. Through the interactive design, students are able to freely adjust the speed, angle and content of the animation according to their own level of understanding, thus achieving a more efficient and personalized learning experience. Together, these advantages significantly enhance teaching and student learning.

6. Conclusion

This paper focuses on graphic design and animation technology for communication optimization, combines key technologies such as bone animation, motion capture, GPU-accelerated rendering and machine learning-assisted design, and proposes a series of optimization methods, which are verified by practical application examples. The main conclusions are as follows:

1) The combination of bone animation and motion capture technology

Using bone animation to drive mesh model deformation and motion capture technology to obtain real motion data, it greatly improves the naturalness and realism of animation. In the applications of Virtual Reality (VR), Augmented Reality (AR) and film and television animation production, the method in this paper effectively solves the problems of stiff and unnatural movements in traditional animation, and the smoothness of animation is increased to 92 points, and the sense of reality is greatly enhanced.

2) The efficiency of GPU-accelerated rendering

Based on GPU parallel computing architecture and optimized rendering pipeline, this method significantly improves the real-time rendering performance of animation. The experimental results show that the proposed method can increase the frame rate from 45 FPS to 60 FPS and reduce the rendering time to 18 ms, which can meet the requirements of high real-time and complex scene rendering and provide technical support for dynamic transmission and interaction.

3) Intelligence and efficiency of machine learning-aided design

By generating adversarial networks (Gans) and deep learning regression models, the proposed method performs well in automatically generating graphic elements and optimizing design quality. The generated graphics have high accuracy and style consistency, which effectively improves the design efficiency and reduces the workload of manual intervention.

4) Application and comparative advantages of the method

Combined with practical application scenarios (such as VR/AR interaction, film and video production and game production), the proposed method is superior to the traditional method in terms of performance indicators (frame rate, rendering time, animation smoothness). Through quantitative analysis and line graph fitting verification, the overall efficiency and animation quality of the proposed method have been significantly improved.

Author contributions: Conceptualization, JX and CS; methodology, DH; software, JX; validation, JX, CS and DH; formal analysis, JX; investigation, CS; resources, DH; data curation, JX; writing—original draft preparation, CS; writing—review and editing, DH; visualization, JX; supervision, CS; project administration, DH; funding acquisition, JX. All authors have read and agreed to the published version of the manuscript.

Ethical approval: Not applicable.

Conflict of interest: The authors declare no conflict of interest.

References

1. Liu X. Animation special effects production method and art color research based on visual communication design. Scientific Programming. 2022; 4: 7835917.
2. Zhao L. The application of graphic language in animation visual guidance system under intelligent environment. Journal of Intelligent Systems. 2022; 31(1): 1037–1054.DOI:10.1515/jisys-2022-0074.
3. Lan H. Research on the strategy of using graphic creativity in visual communication design in the context of Internet. Applied Mathematics and Nonlinear Sciences. 2024; 9(1).
4. Yan H, Jiang H, Wang J, et al. Audience-oriented Aesthetic and Creative Research on Animated Films Based on Visual Communication Design. Applied Mathematics and Nonlinear Sciences. 2023; 9(1).
5. Pan Z, Pan H, Zhang J. The application of graphic language personalized emotion in graphic design. Heliyon. 2024; 10(9).
6. Kathuria A. Design and Optimization of Lighting Techniques for Enhancing Animation Visuals. Nat. Volatiles & Essent. Oils. 2021; 8(3): 192–199.
7. Song Y. Research on the application of computer graphic advertisement design based on a genetic algorithm and TRIZ theory. International Journal of Interactive Multimedia and Artificial Intelligence. 2021; 7(4).
8. Yang X .Application of Visual Communication Design in Digital Media.2023 IEEE International Conference on Integrated Circuits and Communication Systems (ICICACS), 2023:1-6.DOI:10.1109/ICICACS57338.2023.10099784.
9. Setlur V. Optimizing computer imagery for more effective visual communication. Northwestern University; 2005.
10. Cai J, Su J. Application characteristics and innovation of digital technology in visual communication design. Advances in Multimedia. 2022; 17: 1–12.
11. Zhang Y, Su R, Yu J, et al. 3D facial modeling, animation, and rendering for digital humans: A survey. Neurocomputing. 2024; 598: 128168.
12. Dai F, Li Z. Research on 2D Animation Simulation Based on Artificial Intelligence and Biomechanical Modeling. EAI Endorsed Transactions on Pervasive Health and Technology. 2024; 10.
13. Sato Y, Kitazaki M, Itakura S, et al. Great apes' understanding of biomechanics: Eye-tracking experiments using three-dimensional computer-generated animations. Primates. 2021; 62(5): 735–747.
14. Yang Z. 2D animation comic character action generation technology based on biomechanics simulation and artificial intelligence. Molecular & Cellular Biomechanics. 2024; 21(1): 338–338.
15. Mourot L, Hoyet L, Le Clerc F, et al. A survey on deep learning for skeleton-based human animation. Computer Graphics Forum. 2022, 41(1): 122–157.

16. Han J. Application of human-computer interaction technology integrating biomimetic vision system in animation design with a biomechanical perspective. Molecular & Cellular Biomechanics. 2024; 21(4): 468–468.
17. Goswami P. A survey of modeling, rendering and animation of clouds in computer graphics. The Visual Computer. 2021; 37(3–4): 1931–1948.
18. Zeng B, Liu B, Li H, et al. FNeVR: Neural volume rendering for face animation. Advances in Neural Information Processing Systems. 2022; 35: 22451–22462.
19. Chen X, Wang B, Shum HY. Hand avatar: Free-pose hand animation and rendering from monocular video. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2023; 17–24.
20. Ling Y. Design of 3D animation color rendering system supported by cloud computing based on genetic algorithm. Soft Computing. 2023; 27(14): 10317–10326.