Received: 3 Jun. 2024 | Revised: 17 Jul. 2024 | Accepted: 22 Jul. 2024 | Published online: 25 Jul. 2024
RESEARCH ARTICLE

Journal of Global Humanities and Social Sciences
2024,Vol. 5(7)252-257
DOI: 10.61360/BoniGHSS242016590704

# Research on User Experience Optimization of Mobile Application Based on Animation Technology

**Donglan Zou**[1,2] **& Mohamad Yusof Darus**[1,*]
[1] *College of Computing,Informatics and Mathematics, UiTM, Malaysia*
[2] *School of Mathematics and Computer Science, Xinyu University, China*

**Abstract:** With the rapid development of mobile Internet, mobile applications have become an indispensable part of people's daily lives. User experience is an important indicator to measure the quality of the application, which is the overall feeling and experience of the user in the process of interacting with the application. Mobile applications can increase user stickiness and satisfaction if they can create a good user experience. From the technical point of view, the application with quick response and smooth interaction can bring a positive experience for the user, which in turn improves overall satisfaction. From the point of view of the user's actual needs and habits, mobile applications should be designed from a personalization point of view, analyze user behavior and preferences, and carry out all-around consideration and design. Animation technology, as an important part of user interface design, can enhance the visual appeal and user interaction experience, providing a more intuitive and pleasant operating experience. This paper aims to discuss how to optimize the user experience of mobile applications through animation technology.

**Keywords:** animation technology; mobile application; user experience; optimization

## Introduction

Animation techniques are used to create the effect of visually coherent motion by playing a series of static images or frames in succession. The earliest animations were predominantly hand-drawn and represented continuous motion by drawing frame by frame. With the development of technology, animation technology has gradually evolved into computer-aided 2D and 3D animation, which plays an important role in the entertainment field, such as movies, television, and video games, and is also widely used in a variety of industries, such as advertising, education, medicine, and architectural design. The application of animation technology in mobile applications can be traced back to the popularity of smartphones. Initially, animation effects in mobile applications were relatively simple and were mainly used for basic visual feedback, such as color changes when buttons were clicked and simple loading animations. With the improvement of hardware performance and the advancement of development tools, animation effects in mobile applications have become more complex and diverse. In recent years, the application of animation technology has become more and more extensive and in-depth due to the increasing requirements of users for mobile applications. From micro-interactive animation to full-screen animation, from simple loading indicators to complex interactive feedback, animation technology has become an important means to enhance the user experience of mobile applications.

## 1. The Role of Animation Technology in Mobile Applications

### 1.1. Enhance the visual appeal

**Corresponding Author:** Mohamad Yusof Darus
College of Computing,Informatics and Mathematics, UiTM, Malaysia
Email: yusof@fskm.uitm.edu.my

The development of science and technology provides a variety of means of expression for animation, and the development of the network provides a broad display space for animation. In modern mobile application design, animation technology attracts users' attention with vivid visual effects, making the interface more vivid and expressive. Compared with static images and text can convey information and emotions. Smooth transitions and changes guide the user's eyes, highlight important content and functions, and create a high-quality and professional visual experience, so that the user has a positive impression of the overall quality and design level of the application, and improves the recognition and satisfaction of the application (Zhang & Yang, 2018). Animation can also give the application a unique brand personality and visual recognition through the designer's unique creativity and style, so that the application stands out in the competitive market, such as subtle gradient animation, dynamic button clicking effects, and natural page switching animation, all of which allow users to feel the application's meticulousness and attentiveness in the process of using the application, which in turn increases the fondness and loyalty to the application (Su & Xu, 2017).

### 1.2. Enhance user interactive experience

When the user clicks the button, slides the screen, or carries out other operations, the timely animation effect can provide clear feedback on the operation, informing the user that the operation has been accepted and is in the process of execution, which improves the user's operating experience, enhances the responsiveness and vividness of the interface, and also simulates the physical effects to make the interaction of the user interface more natural and intuitive, such as inertial sliding, elasticity of the scrolling and other animation effects, which allows the user to feel the physical behavior in the real world, so that the operation process is more in line with the user's psychological expectations, and enhance the user's comfort and satisfaction (Nong, 2017). In addition, animation can guide the user to complete specific tasks and operational steps to enhance the user's operational efficiency and experience, such as the process of form filling, step-by-step display, and dynamic prompts, the user can more clearly understand the steps and information that need to be completed, reducing operational errors and confusion (Zhu, 2015).

## 2. Flutter Development Framework Introduction

### 2.1. Background and development of Flutter

Flutter is an open-source UI software development toolkit developed by Google. The history of Flutter can be traced back to 2015, known as "Sky", and its main goal is to be able to run applications at 120fps. After several years of development and improvement, Flutter released its official version 1.0 in December 2018, officially stepping into the developer's field of vision. The background of the birth of Flutter is to deal with the existing cross-platform frameworks in the performance and consistency of the shortcomings of the traditional cross-platform frameworks usually rely on the web view or bridging to the local control, which has a lot of limitations in the performance and user experience, Flutter is born in the background to deal with the shortcomings of the existing cross-platform frameworks in the performance and consistency of the existing cross-platform frameworks. Flutter avoids the performance bottleneck of traditional frameworks by using its rendering engine to control the interface at the pixel level.

### 2.2. Features and advantages of Flutter

Flutter is developed in Dart, an object-oriented programming language developed by Google. the high performance and easy-to-learn syntax of Dart, the use of its rendering engine, which does not rely on the platform's native controls, improves the rendering speed and performance but also makes the application on different platforms show a consistent appearance and behavior. Flutter's "hot reload" feature allows developers to see the effect of changes immediately after modifying code, improving development efficiency and iteration speed, especially in the UI debugging and design process. In addition, Flutter provides a rich set of pre-built component libraries, which are not only beautiful but also highly customizable, so that developers can easily adjust and extend them according to their needs. Flutter also has strong cross-platform capabilities, can generate iOS and Android apps through a single codebase, and can even be extended to Web and desktop apps, which reduces development and maintenance costs, and shortens product time-to-market. Flutter's community support is also very strong, with an active developer community and a rich plug-in ecosystem that allows developers to easily find the resources and tools they need to solve problems encountered in the development process.

## 3. Flutter-Based Animation Technology Applications

### 3.1. Basic animation effect realization

Flutter is an important framework for modern mobile application development, one of the core components is the animation controller, which is the cornerstone of the Flutter animation system, and is responsible for controlling the timeline of the animation, that is, the animation duration, playback speed and direction, support for reverse playback and repeat playback functions, to meet the diversified animation needs. Developers can use the animation controller to create frame animation and combine it with a variety of curve functions (Curves) to realize different animation rhythms and effects, such as uniform animation, accelerated animation, decelerated animation bouncing animation, etc., to make the animation performance more vivid and natural. Transition animation (Tween) is used to define the initial state and the end state of the animation, and interpolation is used to generate continuous intermediate states in the animation timeline. The transition animation class is essentially a generic class that supports animation interpolation of multiple data types, including value, color, size position, etc. Transition animation can be combined with animation controllers to achieve smooth and delicate transition effects, such as position animation, Tween<double> can be used to move smoothly from one coordinate to another, calculating the current coordinates in each frame of the animation controller to achieve a smooth movement animation. Similarly, ColorTween can be used to implement color gradient animation, making interface elements transition naturally between different colors to enhance the visual effect. To realize and manage animation effects more conveniently, Flutter provides a variety of animation components, among which, AnimatedBuilder can apply callback functions to rebuild part or all of the subtree of the animation in each frame to realize efficient animation updates; AnimatedWidget makes the animation logic separate from the building logic by inheriting custom animated widgets, which makes the code structure clearer and simplifies the animation. structure is clearer, simplifies the process of realizing animation, and improves the maintainability and readability of the code.

### 3.2. Complex animation effect realization

The realization of complex animation effects is an important means to enhance the user experience and interface interactivity of mobile applications. Multi-animation sequence refers to a set of animations executed in a specific order to form a series of coherent visual effects, which are often used to guide users to complete specific tasks or operational steps. Developers can define multiple animation controllers in Flutter and set different timelines and interpolation methods for each controller, such as triggering the scaling animation of the button after the user clicks the button, executing the color gradient animation, and finally completing the position moving animation. Parallel animation refers to the simultaneous execution of multiple animation effects, such as changing the transparency, position, and size of multiple elements at the same time when the page is switched, launching multiple animation controllers, and associating them with different animation attributes to realize parallel animation effects. Developers can use AnimatedBuilder or AnimatedWidget to integrate multiple animation logics into a single component to realize efficient parallel animation updates, which can simultaneously start the fade animation, position shift animation, and rotation animation of elements when switching pages to form visually rich transition animations. Flutter's high-performance rendering engine Flutter's high-performance rendering engine optimizes the frame rate and rendering efficiency of animations, executing complex animation effects at high frame rates, avoiding common lagging and frame-dropping problems, and ensuring the best visual experience for users.

### 3.3. Use of animation libraries and plugins

Flutter comes with an animation library that provides a variety of built-in animation components and tools, such as 'AnimatedContainer', 'AnimatedOpacity', 'AnimatedPositioned', and other components of Flutter, which encapsulate the complex animation logic and also provide highly customizable attributes that enable developers to Flexible adjustment of animation parameters according to specific needs. In addition, Flutter's animation library contains a large number of predefined animation curves (e.g. 'Curves.easeInOut', 'Curves.bounceInOut', etc.), which are used to control the tempo of the animation and the transition effect and to realize a more natural and vivid animation performance. In addition to its animation libraries, Flutter's ecosystem also has many excellent third-party animation plug-ins, such as the 'flutter_animate' plug-in supports complex animation sequences and parallel animation, and the 'rive' plug-in can easily realize rich visual effects through a simple API. The 'rive' plugin supports vector and skeletal animation, enabling developers to create high-quality animation effects that integrate seamlessly into Flutter apps, and the 'flare_flutter' plugin is suitable for realizing dynamic vector and interactive animations, providing an intuitive editing interface and powerful animation control. When using these third-party plug-ins, developers need to pay attention to compatibility and performance optimization issues with the Flutter framework. When selecting and integrating plug-ins, priority should be given to the stability and performance of the plug-ins and optimization in conjunction with the actual needs.

## 4. Based on the User Experience Optimization of Mobile Application Animation Design Methods

### 4.1. Keep concise and consistent

In mobile application design, animation effects should be simple and clear, avoiding overly complex or fancy designs that may distract the user's attention or increase the cognitive burden, and keeping the animation style consistent throughout the application is equally crucial (Nong, 2017). The theme feature in Flutter allows developers to define global animation parameters such as animation duration and curve type; CSS animation and JavaScript libraries (e.g., Anime.js, GreenSock Animation Platform), can also be integrated into mobile apps via WebView or hybrid development frameworks, adding more options for animation effects.

### 4.2. Improve navigation experience

Enhancing the navigation experience is an important aspect of optimizing the user experience of mobile applications, and guiding the user through the navigation with animation design can make the page switch more natural and smooth, reduce the user's confusion and discomfort, and enhance the user's understanding of the interface structure and hierarchical relationships (Wu, 2015). Hero animation is a special animation effect in Flutter used to create a coherent visual transition when switching between pages, such as image browsing in the application, after clicking the thumbnail, the image can be smoothly expanded from the thumbnail position to the full-screen display through Hero animation, which improves the user's visual experience and operation feeling. The implementation of Hero animation is also relatively simple, just define the Hero component with the same name in two pages and set the corresponding label. In addition, the Animated and LayoutAnimation libraries in React Native, as well as the Lottie animation library, can be integrated into mobile applications through hybrid development, providing more options for navigation animation effects.

### 4.3. Enhancing interactive feedback

Providing instant user feedback animation in mobile application design is an important means of enhancing the interactive experience (Gao, 2011). Core Animation in iOS development provides high-performance graphic rendering and animation capabilities to create smooth and complex animation effects such as scaling, rotation, and panning. UIKit Dynamics, on the other hand, can simulate real physical effects and add vivid animated feedback to user interface elements through properties such as gravity, collision, and elasticity. In Android mobile application development, in addition to standard View animations and property animations, the animation system in Jetpack Compose allows developers to create highly customized animation effects, such as state change animations, transition animations, and multi-step animation sequences to enhance the user interaction experience.

### 4.4. Emphasize key actions

Key operation animations can guide the user's attention and enhance the user's understanding of the application and the accuracy of the operation. Shake Animation is one of the commonly used feedback methods, through a slight and fast shaking effect, the user can intuitively feel which input box needs to be corrected, to improve the efficiency of interaction and user experience (Hu, 2006). ShakeAnimation plugin in Flutter is a powerful tool for realizing Shake Animation, developers can add shaking effect to any component through simple API calls. When form validation fails, use ShakeAnimation to explicitly indicate the input box that needs to be corrected. The AnimatedErrorText component can also be used to display validation error messages, making it easier for users to notice and understand by animating the error alerts. The Animated and LayoutAnimation libraries in React Native provide flexible animation implementations that can add rich animation effects to key operations in mobile applications. The Lottie animation library supports importing animations made by After Effects to create complex vector animations that can be used to emphasize important operations or information.

### 4.5. User-controllable animation

In mobile application design, different users have different levels of acceptance of animation. Some users may prefer rich animation effects for a more vivid visual experience, while others may want to disable animation because of the visual interference it brings or because of device performance issues (Li, 2013). Therefore, providing animation control options in app settings improves user satisfaction and experience. In the Flutter framework, developers can add switch buttons on the app settings page to choose to enable or disable global animation effects, and dynamically adjust the animation parameters in different parts of the app through global state management tools such as Provider or Riverpod. In React Native, developers can use the Context API or Redux to manage the global state and dynamically adjust the behavior of animation components based on user settings. UserDefaults in iOS development and SharedPreferences in Android development can be used to store user preferences and read those preferences when the app is launched to decide whether to enable animation.

## Summary

In summary, from the perspective of user experience, animation technology can provide good visual effects for the interface design and functional design of mobile applications, and thus enhance the user experience. This paper introduces specific strategies to enhance user experience from basic animation effects to the realization of complex animation, and then to the user-controllable animation settings. With the continuous progress of technology and changes in user needs, animation technology will play a more important role in the design of mobile applications and should continue to pay attention to the development trend of animation technology and explore new design methods to achieve a better and personalized user experience effect.

## Conflict of Interest

The author declares that she has no conflicts of interest to this work.

**Acknowledgement**

This research was funded by:

National Natural Science Foundation of China( No. 62062063)

the Science and Technology Research Project of Jiangxi Provincial Department of Education, China (No.GJJ202310);

The Jiangxi Provincial Natural Science Foundation, China (No.20224BAB202022)

## References

Zhang, S., & Yang, Z. (2018). Application status and development trend of interactive animation in mobile three-dimensional user interface. *Popular Literature and Art, 2018*(05), 68–69.

Su, D. N., & Xu, X. R. (2017). Research on interactive animation advertisement based on internet mobile. *Design Art Research, 7*(04), 36–41.

Nong, D. (2017). Application and implementation of mobile client in microclass teaching--taking flash 2D animation production course as an example. *Sichuan Cement, 2017*(04), 312, 295.

Zhu, H. (2015). Exploration of animation original design in mobile applications. *Decoration, 2015*(10), 108–110.

Wu, J. (2015). Application and exploration of animation in mobile devices. *Art Technology, 28*(04), 72.

Gao, H. (2011). Teaching multimedia animation presentations in mobile communication technology course. *Science and Technology Information, 2011*(31), 251.

Hu, H. (2006). Analysis of mobile flash technology development. *Communication World, 2006*(38), 25–27.

Li, M. (2013). Application of three-dimensional animation technology in mobile media interface. *Popular Literature and Art, 2013*(22), 101.

**How to Cite:** Zou, D., & Darus, M. Y. (2024). Research on User Experience Optimization of Mobile Application Based on Animation Technology. *Journal of Global Humanities and Social Sciences*, *05*(07), 252-257. https://doi.org/ 10.61360/BoniGHSS242016590704