MODERN AND CONTEMPORARY RESEARCH IN COMPUTER SCIENCES AND ENGINEERING

All Sciences Academy

MODERN AND CONTEMPORARY RESEARCH IN COMPUTER SCIENCES AND ENGINEERING

Editor
Assoc. Dr. M. Emin ŞAHİN

Modern and Contemporary Research in Computer Sciences and Engineering

Editor: *Assoc. Dr. M. Emin ŞAHİN*

Design: All Sciences Academy Design
Published Date: August 2026
Publisher's Certification Number: 72273
ISBN: 978-625-8839-72-2

www.allsciencesacademy.com
allsciencesacademy@gmail.com

CONTENT

A Multi Functional Mobile Application for Task Management, Mood Analysis, and AI-Based Dream Interpretation

Hakan ÜÇGÜN[1]
Bilge Ceren YILMAZ[2]

1- Asst. of Prof.: Bilecik Şeyh Edebali University Faculty of Engineering Department of Computer Engineering. hakan.ucgun@bilecik.edu.tr ORCID No:0000-0002-9448-0679.
2- Bilecik Şeyh Edebali University Faculty of Engineering Department of Computer Engineering. bilgecerenyilmaz@gmail.com ORCID No: 0009-0002-8839-4835.

ABSTRACT

With the rapid growth of digital technologies, mobile applications supporting time management, psychological well-being, and personal development have become increasingly important. However, most existing solutions address these functions separately rather than through an integrated platform. This book chapter presents the design and development of an Android-based mobile application that combines goal management, psychological assessment, progress tracking, and AI-supported interaction to support personal development. The proposed system is designed for individuals who experience difficulties in organizing daily activities due to forgetfulness, poor planning, or psychological challenges. Firebase is used for secure user authentication and cloud-based data management, while the Gemini API provides AI-driven text analysis and personalized feedback. The mood assessment module evaluates anxiety, obsessive-compulsive symptoms, emotional eating, social phobia, and self-esteem using validated psychological scales, offering AI-assisted interpretations of assessment results. In addition, the AI-powered dream analysis chatbot promotes self-expression and self-awareness through natural language interaction. By integrating digital psychology, AI-based assistance, and daily planning tools into a single platform, the proposed application extends beyond traditional task management systems. The study contributes to the growing field of AI-assisted personal development and digital mental health, providing a practical foundation for future intelligent personal assistant applications.

Keywords – Digital Psychology, Artificial Intelligence, Dream Analysis, Mood Tracking, Smart Personal Assistant

INTRODUCTION

The rapid widespread of mobile technologies and smart devices has led to the emergence of various mobile application solutions to meet individuals' needs for health, psychological well-being, and managing daily life activities. These applications provide a portable and interactive platform that allows users to improve their lifestyles, monitor their emotional state, and organize their behavioral goals. In particular, mobile health (m-health) and personal productivity applications facilitate access to healthcare services by enabling users to continuously monitor their physical and psychological condition.

Current studies show that mobile applications are no longer just passive tools, but have transformed into active digital assistants with user interaction, data analytics, and AI-supported decision-making mechanisms [1, 2]. This

transformation has led to the emergence of new generation systems that directly affect both the cognitive processes and daily habits of individuals. Mobile health applications play a significant role in filling gaps, particularly in mental health services [3]. These applications contribute to individuals' own health management methods by offering alternative and complementary solutions in situations where access to traditional health services is limited [4]. Numerous studies exist in the literature regarding the use of m-Health applications in the fields of mental health and psychological support; these applications can provide additional support to psychotherapy processes, as well as contribute to monitoring and managing an individual's psychological state [5]. However, most existing studies focus on applications targeting a specific function and do not adequately address areas such as personal development, psychological assessment, and AI-supported interaction within an integrated framework.

For individuals to effectively manage their daily lives, multidimensional factors such as time management, mental health, and self-awareness need to be considered together. Time management plays a critical role in task completion success and productivity, while mental state and emotional awareness directly affect an individual's cognitive performance and decision-making processes. Systematic studies conducted in recent years have shown that mobile-based self-monitoring and psychological support applications make significant contributions to reducing anxiety and stress levels [6, 7]. This demonstrates that digital platforms can be used not only as organizational tools but also as psychological support mechanisms.

The use of mobile applications in the field of mental health is becoming increasingly widespread, offering functions such as mood monitoring, stress management, and behavioral analysis. Meta-analysis studies show that mobile-based psychological applications have statistically significant effects on reducing stress levels and improving individuals' psychological well-being [8]. However, studies based on large-scale user data reveal that these applications are particularly effective in individuals with low and moderate symptoms [9]. In this context, digital psychology and AI-supported systems are gaining increasing importance. AI-based mobile applications can provide personalized recommendations by analyzing user data and help individuals understand their psychological state. In particular, the integration of natural language processing and machine learning techniques makes it possible to

analyze user inputs contextually and allows for the development of more interactive systems [10, 11].

Studies in the literature reveal that a large proportion of mobile mental health applications are of low quality and are not developed based on scientific principles [9]. Furthermore, the presence of thousands of different applications in the application ecosystem makes it difficult for users to choose the right and reliable application [12]. This situation constitutes a significant problem in terms of user experience and sustainable use. When current mobile applications are examined, it is seen that the vast majority offer limited solutions focusing on a single problem. Task management applications support users' daily planning, while mood tracking applications focus only on monitoring emotional states. Similarly, AI-powered mental health applications generally serve within a specific functional framework. Current studies reveal that these applications have various shortcomings in terms of usability, data integrity, and integration [13, 14]. This fragmented structure leads users to use multiple applications for different needs, and a holistic user experience cannot be offered. This situation stands out as a significant gap in the literature.

In this book chapter presents the design and development of a multifunctional mobile personal development application. The study offers an integrated system that combines components enabling users to plan daily goals, assess psychological mood, and utilize AI-assisted interpretation. This approach aims not only to focus on tracking behavioral goals but also to generate data on an individual's psychological state and transform this data into self-awareness. Furthermore, the AI-assisted analysis component allows users to receive contextual feedback through free-text input (e.g., dream sharing). In this context, the study addresses gaps in the existing literature by: combining mobile applications on a multi-functional basis, integrating psychological assessment and behavioral goal tracking on the same platform, and implementing AI-assisted contextual analysis mechanisms. Thus, it proposes a solution that holistically addresses user needs and offers a unique contribution to the research fields of mobile psychology, personal development, and AI applications.

RELATED WORKS

Developments in mobile technologies, digital psychology and AI have led to radical transformations in how individuals manage their daily lives. In this context, mobile applications stand out as versatile tools supporting task management, mental health monitoring, AI-assisted interaction, and self-reflection processes. In particular, m-health applications have become a significant research focus in the field of mental health in recent years. These applications allow individuals to monitor their psychological state, analyze their behavioral patterns, and manage their personal health more effectively. With the widespread use of smartphones, mobile-based solutions have become an integral part of daily life, and various studies have shown that they offer significant advantages in early intervention processes [7, 15].

The integration of artificial intelligence into mobile applications has made it possible to develop more personalized, scalable, and data-driven solutions. In this context, a comprehensive review conducted by Milne-Ives et al. [16] reveals that AI and machine learning techniques are widely used in mental health applications, and supervised learning models are particularly preferred in mood prediction and behavior analysis processes. However, the study highlights that many applications fall short in terms of clinical validation and long-term impact assessment. Huang et al. [17] adopted a mixed-methods approach by addressing the integration of mobile applications, artificial intelligence, and teletherapy systems; combining user surveys with system-level analyses. The findings show that hybrid systems combining synchronous and asynchronous intervention methods increase accessibility and positively affect treatment adherence. In addition, it is seen that generative artificial intelligence and large language models can produce personalized outputs by contextually analyzing user data. In the study by Ni and Jia [18], it was shown that AI-assisted digital intervention systems are effective in managing anxiety and depression. Some studies in the literature also support the idea that AI-based mobile applications increase user interaction and offer more dynamic analysis possibilities [19, 20].

A significant portion of mobile mental health applications focus on mood tracking. These applications allow users to collect daily mood data and analyze changes over time. A study by Tseng et al. [21] showed that mood data obtained through smartphone-based systems had significant relationships

with behavioral patterns. In this study, both passive perception data and self-reports were used together to analyze mood changes; the integration of sensory data such as activity and sleep patterns revealed a strong correlation between behavioral signals and mood. Similarly, a mobile health tracking system developed by Polhemus et al. [22] addressed depression management using longitudinal data collection and statistical modeling methods and showed that regular self-reports contributed to the early detection of depressive symptoms. However, it appears that a large portion of existing studies are limited to data collection and visualization, and there are significant shortcomings in the in-depth analysis of the obtained data and its transformation into personalized feedback.

User interaction and sustainability in mobile health applications are also emerging as important research areas. Giovanelli et al. [23] examined user interactions in mental health applications using behavioral analysis methods and revealed that interactive features have positive effects on long-term use. The Appa Health platform developed within the scope of this study offers a structured mentoring system for young individuals showing symptoms of depression and anxiety, and the applicability and user acceptance of the system were evaluated. The findings clearly demonstrate the importance of user-oriented design approaches and adaptive systems. In addition, in the study conducted by Chou et al. [24] on elderly users, a usability-oriented mobile health system was proposed, and it was shown that simplified interface designs significantly increased the adoption rate by elderly individuals.

Chatbot-based mobile mental health applications have become a significant area of research in recent years due to their potential to increase user interaction and expand accessibility. Haque and Rubya [25] evaluated chatbot-based systems through content analysis based on descriptions and user reviews in app stores. The study's findings reveal that user experience largely depends on the empathy level and interaction quality of chatbots. Furthermore, it is emphasized that these systems offer significant advantages in terms of accessibility, but need improvement in personalization and reliability. Similarly, Jabir et al. [26] systematically reviewed conversational agents used in digital mental health applications and showed that integrating chatbots with behavioral modification techniques increased user interaction. However, the limited clinical validation processes of these systems were noted as a significant shortcoming.

Yang et al. [11], in their review of AI-based mobile mental health applications for children, stated that machine learning and natural language processing techniques play an effective role in obtaining meaningful inferences from user data. The findings show that these systems provide significant advantages in terms of offering personalized recommendations. On the other hand, Aboujaoude et al. [27], in their study evaluating the current state of digital mental health interventions, revealed that these systems are mostly supported by cognitive behavioral therapy (CBT) based approaches. Based on the analysis of randomized controlled trials and clinical studies, this research emphasizes that digital interventions can be effective, but user engagement is a critical factor in terms of sustainability. Shen et al. [28] examined mental health monitoring systems by analyzing sensor data, user interactions, and environmental factors together within the framework of a behavioral sensing approach. The study results show that the integration of multiple data sources provides more accurate and reliable predictions. These findings indicate that systems based solely on self-reports may be limited and that more comprehensive data integration is needed.

In the current literature, most mobile health and psychological support applications focus on singular functions such as task management, mood tracking, or daily logging, failing to address the multifaceted life dynamics of individuals holistically. Furthermore, these applications generally only collect user data but are insufficient in interpreting this data to generate personalized and contextual feedback. The integration of artificial intelligence technologies into mobile applications is mostly limited and not used to provide in-depth psychological analysis and behavioral modeling. However, issues such as flexible task performance evaluation, linking dream analysis to psychological states, and integrating multiple data sources are not adequately addressed in the literature, creating a significant research gap.

This study presents an innovative mobile application approach that integrates task management, mood analysis, and AI-assisted dream interpretation functions into a single platform to address the gaps in the literature. The proposed system provides more accurate user performance analysis using a percentage-based evaluation model instead of the classic dual-task completion structure; it performs in-depth mood analysis through structured psychological tests; and it generates personalized feedback using generative AI technologies. Furthermore, the integration of the dream analysis

module with other system components allows for the simultaneous evaluation of an individual's cognitive and psychological processes. In these respects, the study offers a unique and comprehensive contribution to the literature in terms of multidimensional data analysis and integrated system design.

METHODOLOGY

The proposed system is designed as a multi-layered and modular mobile application architecture, enabling seamless interaction between users, cloud services, and artificial intelligence-based analysis mechanisms. As illustrated in Figure 1, the system is structured into three primary layers: the Mobile Client Layer, the Cloud Data Management Layer, and the Processing and Analysis Layer.

The Mobile Client Layer serves as the main interface through which users interact with the system. It is responsible for collecting user inputs, including task entries, psychological assessment responses, and dream narratives. This layer is implemented using the Kotlin programming language within the Android Studio development environment, ensuring a responsive and user-friendly interface.

The Cloud Data Management Layer manages secure data storage, synchronization, and user authentication processes. This layer utilizes Firebase services, including Authentication, Realtime Database, and Firestore, to ensure scalable and real-time data handling. All user-generated data is transmitted securely to the cloud infrastructure, where it is stored and made accessible for further processing.

The Processing and Analysis Layer constitutes the core intelligence of the system. It processes the collected data and generates meaningful insights using rule-based methods and artificial intelligence techniques. This layer integrates multiple analytical modules, enabling comprehensive evaluation of user behavior and psychological states.

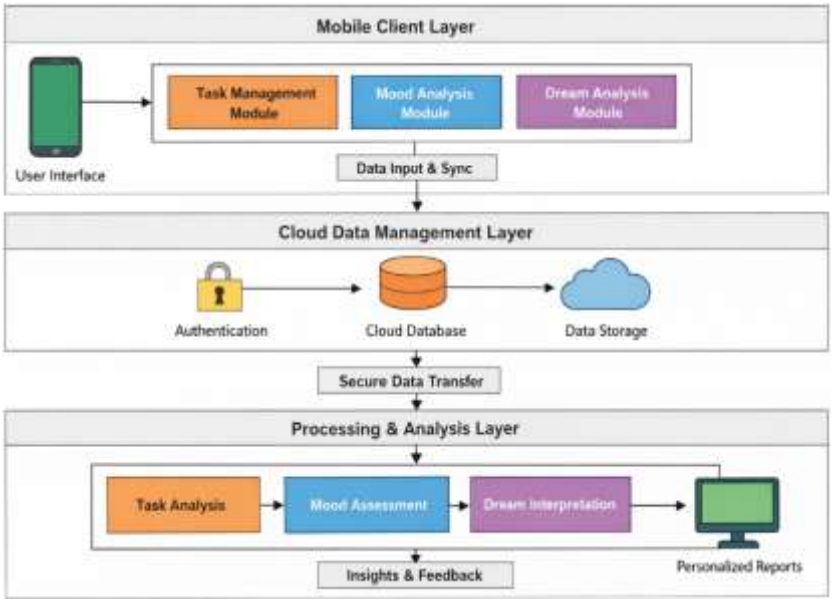


Figure 1. Overview of the Proposed System Architecture

As depicted in Figure 1, the system includes three main functional modules that operate within the processing layer:

• The Task Management Module enables users to define, organize, and monitor their daily activities. Unlike traditional task management systems that rely on binary completion states, this module introduces a percentage-based completion mechanism. This approach allows users to express partial progress, thereby generating more granular behavioral data. The collected task-related data is analyzed over time to identify user habits, productivity patterns, and consistency levels. This provides a foundation for adaptive feedback mechanisms and supports long-term behavioral insights.

• The Mood Analysis Module is designed to assess users' psychological conditions through structured evaluation mechanisms. Instead of relying solely on subjective mood inputs, the system incorporates multiple standardized assessment categories, including anxiety, obsessive-compulsive tendencies, social phobia, emotional eating, and self-confidence. User responses are processed using predefined scoring algorithms to generate quantitative results. These results are further interpreted to provide qualitative feedback. This dual-layer analysis enhances the depth and reliability of psychological evaluation compared to conventional mood tracking applications.

• The Dream Analysis Module utilizes artificial intelligence techniques to interpret user-provided dream narratives. Users input textual descriptions of their dreams, which are then preprocessed and transmitted to an AI-based analysis engine. The system employs a generative AI model to produce context-aware interpretations based on the semantic structure of the input text. Unlike traditional keyword-based systems, this approach enables dynamic, personalized, and context-sensitive analysis. The output is presented to the user as an interpretive narrative, enhancing self-awareness and cognitive reflection.

SOFTWARE ARCHITECTURE AND AI INTEGRATION

This section presents a detailed and systematic account of the technological infrastructure, data collection and processing processes, artificial intelligence modeling approach, and experimental setup used in the development of the MyDailyMate system. The proposed methodology creates a multi-layered and integrated structure through the integration of mobile software development, cloud computing, and artificial intelligence techniques. The MyDailyMate system is designed in accordance with Android-based mobile application development principles and implemented using the Kotlin programming language. The Material Design approach has been adopted in the user interface design, creating a simple and intuitive interface structure to increase user interaction. This approach facilitates user interaction with the system and increases the usability of the application. Cloud-based solutions have been preferred in the system architecture, and Firebase infrastructure has been integrated within this scope. Firebase Authentication securely manages user authentication processes, while Realtime Database and Cloud Firestore services perform data storage and synchronization operations. Thanks to this structure, the system can operate with high availability and low latency. Artificial intelligence-based analysis processes are carried out via the Gemini API, which provides access to large language models. This integration enhances the system's natural language processing capabilities and allows for contextual analysis of textual data obtained from the user. Figure 2 shows the technological components used within the MyDailyMate application.

Figure 2. MyDailyMate Technological Infrastructure System and Components

MyDailyMate adopts a multidimensional data collection approach based on user interactions. The data obtained within the system is classified into four main categories: task management data, mood assessment results, dream texts, and AI analysis outputs. Task data consists of time series data including users' daily activities and their completion rates. Mood data includes numerical scores obtained through psychometric tests and quantitatively expresses the individual's psychological state. Dream data is entered by the user in free text format and analyzed using natural language processing techniques. The data structure is organized with a document-based model on Cloud Firestore. Data for each user is stored with a timestamp, enabling temporal analysis of user behavior. This structure protects the system's data integrity while also increasing its scalability. Figure 3 presents the data flow diagram of the MyDailyMate application.

Figure 3. MyDailyMate Data Flow Diagram

The data processing process in the system is carried out through a multi-stage pipeline structure. This process involves subjecting the data received from the user to specific operations before analysis. In the first stage, the data obtained from the user is collected by the system and undergoes a validation process. Then, in the preprocessing stage, incomplete or erroneous data is filtered and made suitable for analysis. While tokenization, stop-word removal, and basic semantic analysis operations are applied to text-based data; numerical data is normalized and transformed into a comparable form. In the analysis stage, the data is evaluated using an artificial intelligence model, and personalized feedback is generated for the user. This process operates in real-time and continuously updates the user experience. Figure 4 presents the data processing process of the MyDailyMate application.

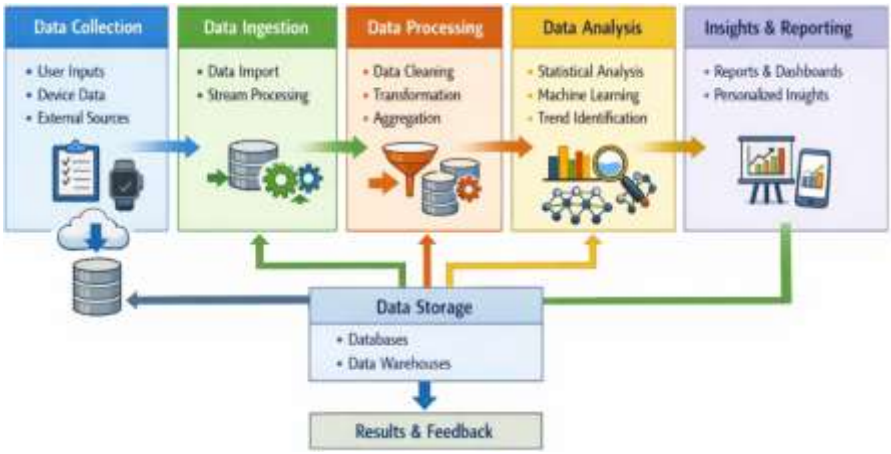


Figure 4. MyDailyMate Data Processing Flowchart

The artificial intelligence model used in the MyDailyMate system is based on large language models and offers high accuracy and contextual understanding in natural language processing tasks. This model analyzes textual data obtained from the user and produces meaningful and personalized outputs. A prompt engineering approach has been applied to improve system performance. Prompts that generate psychological assessments and guiding suggestions in mood analysis have been preferred. This structured approach enables the AI model to produce more consistent and contextually accurate outputs. It also makes a significant contribution to personalizing and making

the feedback provided to the user meaningful. Figure 5 shows the AI integration process within the MyDailyMate application.

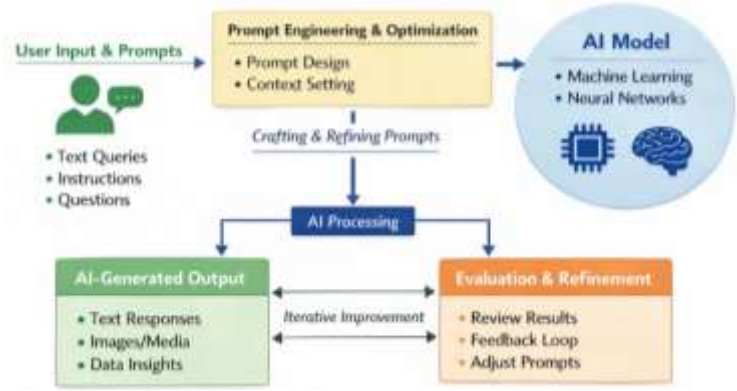


Figure 5. MyDailyMate AI Integration Process

A comprehensive experimental setup was conducted to evaluate the system's performance. This involved creating different user scenarios and analyzing how the system performs under various conditions. Functional tests verified the correct operation of the system modules, while performance tests measured the system's response times and data processing capacity. User experience tests analyzed user interactions to assess the system's usability. Furthermore, the accuracy and consistency of the AI outputs were examined according to specific criteria. As a result of these tests, the system demonstrated successful performance in both technical and user experience aspects. Figure 6 presents the test process flowchart for the MyDailyMate application.

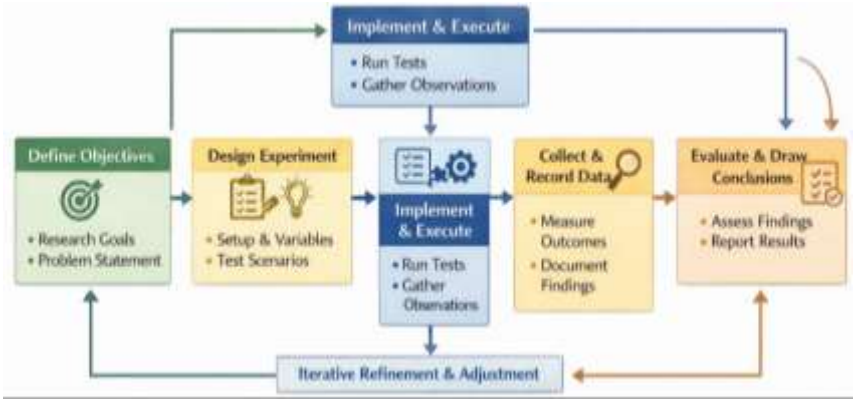


Figure 6. MyDailyMate Test Process Flowchart

USER INTERFACE DESIGN AND IMPLEMENTATION

The user interface (UI) of the proposed mobile application has been designed with a user-centered approach, aiming to ensure usability, accessibility, and seamless interaction across all functional modules. The interface structure follows a modular design paradigm that aligns with the system architecture presented in Figure 1, enabling users to efficiently navigate between task management, mood analysis, and dream interpretation functionalities. Particular emphasis has been placed on simplicity and clarity in order to reduce cognitive load and improve user engagement.

The user authentication interface is designed to ensure system security and enable user-specific data management. This interface plays a critical role as the starting point of the usage process, while also providing users with secure access to the system. A simple and intuitive design has been preferred in the authentication process, aiming to allow users to log in with minimal steps. This approach improves the user experience and increases the system's accessibility. Furthermore, this structure, designed with a balance between security and usability, reinforces user trust in the system. Figure 7.a shows the login interface and 7.b shows the registration interface.

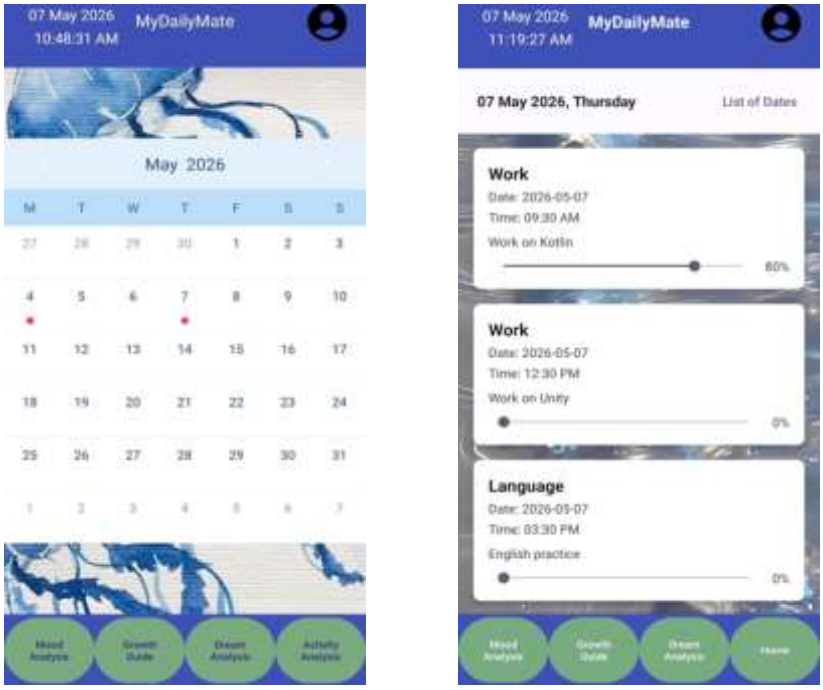
(a) (b)

Figure 7. Login and registration interface

The main control panel is designed as the central control point of the MyDailyMate system. This screen features a minimalist and calendar-focused design. The monthly calendar component, centrally located, visualizes the days users entered data or interacted with the system using colored markers, making historical data tracking effective. The navigation menu integrated into the lower section of the screen provides direct access to the system's core analytics and guidance functions. This modular structure allows users to quickly upload their daily data to the system and access personal development tools seamlessly. Figure 8.a shows the main dashboard interface.

The Task Management interface is one of the core modules that allows users to define their daily goals and organize their activities. This screen enables users to define their tasks in a structured manner. During the design process, a simple and focused interface was created to support users' time management skills. This structure contributes to individuals developing planning habits and adopting a more disciplined lifestyle. Through this screen, users can systematically organize their plans and track their task completion processes. One of the prominent features of the interface is the percentage-based progress bar integrated into each task card. This quantitative and visual completion mechanism allows users to instantly assess their current status in the task processes. In addition to visual indicators representing completion levels, the interface provides a structured list view of tasks. Figure 8.b shows the Task Management interface.

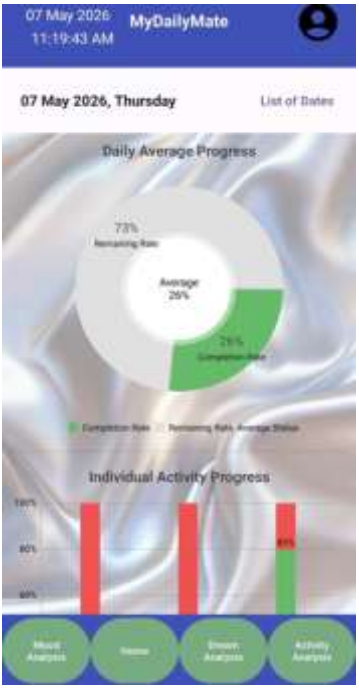
The Progress Screen allows users to track their progress based on the activities they perform and the data they obtain within the system. This screen has a structure where user data is presented in a visual format and individual development is analyzed. Data presented through graphs helps users understand their changes over time. This approach facilitates users' self-assessment of their behavior and increases their level of awareness. As an important component supporting the "self-monitoring" process, the Progress Screen contributes to users making informed decisions. Figure 9.a shows the Progress Screen interface.



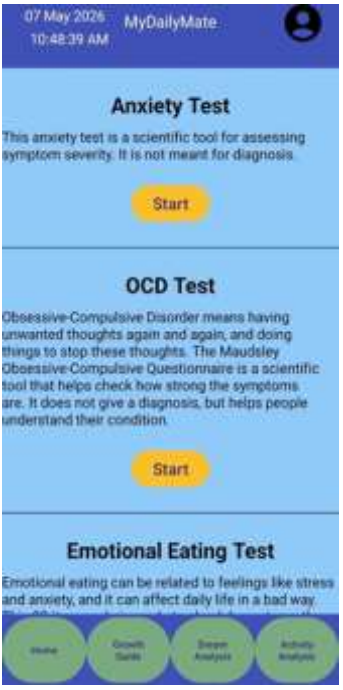
(a) (b)

Figure 8. Main dashboard and Task management interfaces

The Mood Analysis interface provides a structured interface that allows users to assess their mood and psychological state. The interface offers structured questionnaires related to anxiety, social behavior, emotional patterns, and self-esteem. This screen presents scientifically based psychometric scales in a user-friendly format. User input is collected through interactive elements such as multiple-choice questions and rating scales. Users can obtain quantitative results regarding their psychological state based on their answers to specific questions. This data is analyzed by the system, providing meaningful feedback to the user. This process helps individuals become aware of and manage their emotional states. Furthermore, with regular use, users' psychological changes can be monitored over time. Figure 9.b shows the Mood Analysis interface.



(a)



(b)

Figure 9. Progress Screen and Mood Analysis interfaces

The dream analysis screen is designed as an interactive module where users can express their dreams textually, and this data is analyzed by artificial intelligence. This screen provides two-way communication between the user and the system, allowing for contextual interpretation of dream content. The AI-powered analysis process not only presents results but also provides explanatory and guiding feedback to the user. This interactive structure allows users to evaluate their inner experiences more deeply and increase their psychological awareness. The design focuses on simplicity and provides a text input field where users can freely describe their experiences. After submission, the system processes the input and returns an interpretation generated by artificial intelligence. The results are displayed in a structured and readable format that enhances user participation and interpretability. Figure 10 shows the Dream Analysis interface.

The Results and Feedback interface presents the outputs generated by the system, including task performance summaries, psychological analysis results, and dream interpretations. This interface integrates data from multiple modules and provides personalized insights in a cohesive manner. Visual

elements such as progress indicators and categorized feedback enhance readability and user comprehension.

Figure 10. Dream analysis interface

CONCLUSION

This study presents MyDailyMate, a system developed to enable individuals to manage their daily lives more effectively. The proposed system combines mobile application technologies, digital psychology approaches, and artificial intelligence-based analysis mechanisms to provide a multidisciplinary and integrated solution. The integration of different modules such as task management, mood analysis, progress tracking, and AI-assisted dream analysis under a single platform distinguishes the system from existing solutions in the literature.

The system architecture developed within the scope of this study offers a sustainable solution in terms of both technical and application aspects, thanks to its modular and scalable structure. The data management mechanism, supported by a Firebase-based cloud infrastructure, enables the secure and real-time processing of user data. Furthermore, AI-powered

analysis processes implemented through Gemini API integration allow for contextual evaluation of user input and the generation of personalized feedback.

Experimental results demonstrate that the system exhibits strong technical performance with low latency, high availability rate, and efficient data processing capacity. Furthermore, user interaction analyses reveal that the AI-powered modules, in particular, increase user engagement and encourage active system usage. These findings indicate that the proposed system offers a valuable contribution not only from a technical standpoint but also in terms of user experience and personal development.

One of the most important contributions of this study is that it presents task management, psychological assessment, and AI-assisted analysis processes, which are often discussed separately in the literature, under an integrated structure. This approach allows for the multidimensional analysis of user behavior and enables more meaningful inferences to be obtained. Furthermore, the integration of AI technologies not only for a single function but for all components of the system strengthens the innovative aspect of the study.

However, the study has some limitations. First, the system was evaluated using a limited user scenario and dataset. Studies with larger and more diverse user groups will increase the generalizability of the system. Furthermore, the outputs of the AI model largely depend on the accuracy of the data provided by the user. This can lead to variability in analysis results, especially with free-text input.

Future studies propose several suggestions for further developing the system. Firstly, enhancing the AI model with more advanced personalization capabilities will improve the quality of feedback provided to the user. Secondly, integrating biometric sensor data (e.g., heart rate, sleep data) into the system will allow for more comprehensive user analysis. Furthermore, expanding the system to different platforms (iOS, web-based applications) will increase user reach. Additionally, expanding data collection processes for analyzing long-term user behavior will enhance the system's learning capacity. Developing AI-powered recommendation systems and providing proactive suggestions to the user will further improve the system's interaction level. Finally, integrating clinical-level psychological assessment tools is

considered a significant research direction that could increase the system's potential in the digital health field.

In conclusion, the MyDailyMate system offers an innovative approach at the intersection of mobile application, artificial intelligence, and digital psychology, and stands out as a comprehensive digital platform supporting individuals' personal development processes. The proposed structure serves as an important reference for future smart life support systems.

REFERENCE

[1] R. Jeya, Hezerul Abdul Karim, & Sarina Binti Mansor. (2024). Artificial Intelligence and Mobile Apps Support Intelligent Healthcare Systems for Mental Health Services. International Journal of Interactive Mobile Technologies (iJIM), 18(20), pp. 157–168. https://doi.org/10.3991/ijim.v18i20.50743.

[2] Alduhailan, H. W., Alshamari, M. A., & Wahsheh, H. A. M. (2025). A Comprehensive Comparison and Evaluation of AI-Powered Healthcare Mobile Applications' Usability. Healthcare, 13(15), 1829. https://doi.org/10.3390/healthcare13151829.

[3] Yonas Deressa Guracho, Susan J. Thomas, Khin Than Win, Smartphone application use patterns for mental health disorders: A systematic literature review and meta-analysis, International Journal of Medical Informatics, Volume 179, 2023, 105217, https://doi.org/10.1016/j.ijmedinf.2023.105217.

[4] Choudhury A, Kuehn A, Shamszare H, Shahsavar Y. Analysis of Mobile App-Based Mental Health Solutions for College Students: A Rapid Review. Healthcare (Basel). 2023 Jan 16; 11(2):272. https://doi.org/10.3390/healthcare11020272.

[5] F. Kabadayı and M. Güven, "Psikolojik Yardım Hizmetlerindeki Mobil Uygulamalarla İlgili Bir İnceleme," Türk Eğitim Bilimleri Dergisi, vol. 21, no. 3, pp. 1660–1689, Dec. 2023, https://doi.org/10.37217/tebd.1291285.

[6] Askarizadeh, M.M., Gholamhosseini, L., Khajouei, R. et al. Determining the impact of mobile-based self-care applications on reducing anxiety in healthcare providers: a systematic review. BMC Med Inform Decis Mak 25, 37 (2025). https://doi.org/10.1186/s12911-024-02817-4.

[7] Almuqrin A, Hammoud R, Terbagou I, Tognin S, Mechelli A. Smartphone apps for mental health: systematic review of the literature and five recommendations for clinical translation. BMJ Open. 2025 Feb 11;15(2):e093932. https://doi.org/10.1136/bmjopen-2024-093932.

[8] Linardon, J., Firth, J., Torous, J., Messer, M., & Fuller-Tyszkiewicz, M. (2024). Efficacy of mental health smartphone apps on stress levels: a meta-analysis of randomised controlled trials. Health Psychology Review, 18(4), 839–852. https://doi.org/10.1080/17437199.2024.2379784.

[9] Fürtjes, S., Gebel, E., Kische, H. et al. Characteristic of mental health app usage: a cross-sectional survey in the general population. BMC Public Health 24, 3133 (2024). https://doi.org/10.1186/s12889-024-20500-1.

[10] García, A., Utilizing Artificial Intelligence and Mobile Applications for Mental Healthcare: A Social Informatics Viewpoint, Journal of AI-Powered Medical Innovations, Volume 1, Issue 1, 43-57, 2024.
[11] Yang F, Wei J, Zhao X, An R, Artificial Intelligence–Based Mobile Phone Apps for Child Mental Health: Comprehensive Review and Content Analysis, JMIR Mhealth Uhealth 2025;13:e58597 https://doi.org/10.2196/58597.
[12] King, D.R., Emerson, M.R., Tartaglia, J. et al. Methods for Navigating the Mobile Mental Health App Landscape for Clinical Use. Curr Treat Options Psych 10, 72–86 (2023). https://doi.org/10.1007/s40501-023-00288-4.
[13] Alduhailan, H. W., Alshamari, M. A., & Wahsheh, H. A. M. (2025). A Comprehensive Comparison and Evaluation of AI-Powered Healthcare Mobile Applications' Usability. Healthcare, 13(15), 1829. https://doi.org/10.3390/healthcare13151829.
[14] Y. Inal, C. E. Reme, A. R. Blasiak et al., "Usability Evaluations of Mobile Mental Health Technologies: Systematic Review," Journal of Medical Internet Research, 22(1):e15337. 2020. https://doi.org/10.2196/15337.
[15] Chiu YH, Lee YF, Lin HL, Cheng LC, Exploring the Role of Mobile Apps for Insomnia in Depression: Systematic Review, J Med Internet Res 2024;26:e51110, doi: 10.2196/51110.
[16] Milne-Ives M, Selby E, Inkster B, Lam C, Meinert E (2022) Artificial intelligence and machine learning in mobile apps for mental health: A scoping review. PLOS Digit Health 1(8): e0000079. https://doi.org/10.1371/journal.pdig.0000079.
[17] Huang W, Xing Y, Zhao F and Wang Y (2026) Mobile apps, AI, and teletherapy: a comprehensive review of digital mental health tools for nurses. Front. Public Health 13:1686766. doi: 10.3389/fpubh.2025.1686766.
[18] Ni, Y., & Jia, F. (2025). A Scoping Review of AI-Driven Digital Interventions in Mental Health Care: Mapping Applications Across Screening, Support, Monitoring, Prevention, and Clinical Education. Healthcare, 13(10), 1205. https://doi.org/10.3390/healthcare13101205.
[19] N. Bensalah, H. Ayad, A. Adib and A. I. E. Farouk, "MindWave app: Leveraging AI for Mental Health Support in English and Arabic," 2024 IEEE 12th International Symposium on Signal, Image, Video and Communications (ISIVC), Marrakech, Morocco, 2024, pp. 1-6, doi: 10.1109/ISIVC61350.2024.10577908.
[20] S. Raj, S. Shekhar and B. P, "MediHealth: An AI-Driven Mobile Solution for Enhanced Health Care Decision-Making and Accessibility," 2025 3rd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT), Bengaluru, India, 2025, pp. 874-880, doi: 10.1109/IDCIOT64235.2025.10915087.
[21] Yu-Ching Tseng, Esther Ching-lan Lin, Chung Hsien Wu, Huei-Lin Huang, Po See Chen, Associations among smartphone app-based measurements of mood, sleep and activity in bipolar disorder, Psychiatry Research, Volume 310, 2022, 114425, https://doi.org/10.1016/j.psychres.2022.114425.
[22] Polhemus A, Simblett S, Dawe-Lane E, Gilpin G, Elliott B, Jilka S, Novak J, Nica RI, Temesi G, Wykes T, Health Tracking via Mobile Apps for Depression Self-management: Qualitative, Content Analysis of User Reviews, JMIR Hum Factors 2022;9(4):e40133, doi: 10.2196/40133.

[23] Giovanelli A, Sanchez Karver T, Roundfield KD, Woodruff S, Wierzba C, Wolny J, Kaufman MR, The Appa Health App for Youth Mental Health: Development and Usability Study, JMIR Form Res 2023;7:e49998, doi: 10.2196/49998.

[24] Chou YH, Lin C, Lee SH, Chang Chien YW, Cheng LC. Potential Mobile Health Applications for Improving the Mental Health of the Elderly: A Systematic Review. Clin Interv Aging. 2023;18:1523-1534, https://doi.org/10.2147/CIA.S410396.

[25] Haque MDR, Rubya S, An Overview of Chatbot-Based Mobile Mental Health Apps: Insights From App Description and User Reviews, JMIR Mhealth Uhealth 2023;11:e44838, doi: 10.2196/44838.

[26] Jabir AI, Martinengo L, Lin X, Torous J, Subramaniam M, Tudor Car L, Evaluating Conversational Agents for Mental Health: Scoping Review of Outcomes and Outcome Measurement Instruments, J Med Internet Res 2023;25:e44548, doi: 10.2196/44548.

[27] Aboujaoude E, Gega L, Parish MB and Hilty DM (2020) Editorial: Digital Interventions in Mental Health: Current Status and Future Directions. Front. Psychiatry 11:111. doi: 10.3389/fpsyt.2020.00111.

[28] Shen S, Qi W, Zeng J, Li S, Liu X, Zhu X, Dong C, Wang B, Shi Y, Yao J, Wang B, Lou X, Gu S, Li P, Wang J, Jiang G, Cao S, Passive Sensing for Mental Health Monitoring Using Machine Learning With Wearables and Smartphones: Scoping Review, J Med Internet Res 2025;27:e77066, doi: 10.2196/77066.

AI Integration in Mobile Learning: Android-Based Exam and Student Tracking System Application

Hakan ÜÇGÜN[1]
Kardelen ERTE[2]

1- Asst. of Prof.: Bilecik Şeyh Edebali University Faculty of Engineering Department of Computer Engineering. hakan.ucgun@bilecik.edu.tr ORCID No:0000-0002-9448-0679.
2- Bilecik Şeyh Edebali University Faculty of Engineering Department of Computer Engineering. ertekardelen@gmail.com ORCID No: 0009-0008-7785-8108.

ABSTRACT

In recent years, the rapid growth of mobile technologies and cloud infrastructures has transformed digital learning environments. Mobile learning systems have evolved beyond content delivery to monitor learner behavior, analyze performance, and provide adaptive feedback. Consequently, the integration of artificial intelligence (AI) into mobile examination and student monitoring systems has become an important research area in educational technologies. This book chapter presents an Android-based, AI-supported examination and student monitoring system designed to support students preparing for university entrance exams. The proposed system adopts a modular architecture integrating a Flutter-based mobile application, Firebase cloud services for authentication and real-time data management, and an AI assistant powered by the Gemini API for question answering, learning support, and motivational guidance. The application includes topic-based learning materials, video lessons, test-solving modules, mock exam tracking, performance analysis, progress visualization, and AI-driven interaction. By enabling personalized performance evaluation and continuous learning support, the proposed architecture offers a scalable engineering solution that combines software engineering principles with modern educational technologies.

Keywords – Mobile Learning, Student Monitoring System, Artificial İntelligence, Android Application, Digital Examination System

INTRODUCTION

21st-century educational environments are undergoing a radical transformation with digitalization and mobile technologies. Rapid advancements in information and communication technologies are fundamentally transforming the structure of education systems and learning processes. The widespread use of mobile devices, in particular, has made learning independent of time and place, paving the way for the emergence of new learning paradigms as alternatives to traditional educational approaches. In this context, mobile learning (m-learning) stands out as an innovative approach offering significant advantages such as accessibility, flexibility, and personalization in education [1, 2]. M-learning is an interactive and learner-centered pedagogical approach that allows learners to access educational materials from anywhere and at any time during the educational process. By digitizing learning processes, m-learning provides an environment that supports accessibility, flexibility, and the ability of students to manage their

own learning process, and therefore has become an important research topic in the field of educational technologies [3, 4]. In the literature, m-learning is defined as a comprehensive framework that allows students to access course materials, manage learning activities, and track performance data through mobile devices. Systematic studies have shown that mobile learning environments increase learner interaction and positively affect learning performance [5].

Systematic studies examining the adoption and effective use of m-learning have shown that student motivation, ease of use, learning automation, and technological infrastructure are critical factors in the success of applications [3]. In particular, the use of mobile learning platforms in higher education positively impacts student performance and learning behaviors by enabling continuous access to educational materials for students [4]. M-learning allows individuals to manage their own learning processes by providing access to learning content from anywhere via wireless communication technologies and portable devices. This plays a critical role, especially in developing individualized learning experiences and supporting the concept of lifelong learning [6]. Furthermore, recent meta-analysis studies show that m-learning applications have significant and positive effects on students' academic achievement [7]. Moreover, the literature emphasizes that mobile learning has the potential to improve students' critical thinking skills [8]. In this context, monitoring student performance plays a critical role in both improving learning outcomes and customizing individual learning paths.

Artificial intelligence technologies are offering new opportunities in education systems by integrating into the personalization of education, automated assessment, intelligent content recommendations, user behavior analysis, and learning analytics [9, 10]. AI-powered mobile learning environments can create personalized learning paths and optimize learning processes by analyzing student profiles and interaction history [11]. This trend is considered a central component of adaptive learning systems and learning analytics, as AI algorithms offer powerful tools in predicting educational outcomes and early detection of learning difficulties [12]. AI-powered adaptive learning systems optimize learning processes by providing dynamic content based on students' individual characteristics [10]. Such systems can provide personalized recommendations and adaptive quizzes by analyzing

learners' ability levels and learning speeds, which is a significant difference from traditional learning models.

The combination of mobile application development environments and cloud-based infrastructures creates engineering requirements such as secure storage of educational data, real-time synchronization, and user authentication. For example, cloud platforms like Firebase effectively provide critical functions such as data synchronization and performance tracking in mobile student tracking systems [13]. At the same time, AI-supported educational components can be integrated into applications as intelligent assistants designed to support sustainable student performance in educational contexts and increase user interaction [14]. In recent years, the use of mobile-based learning applications has increased significantly, especially in higher education and exam-oriented education systems. Interactive content, instant feedback mechanisms, and data-driven learning analytics offered by mobile technologies increase students' active participation in learning processes and make the learning experience more efficient [15]. However, a large portion of existing mobile learning applications focus on specific functions and are insufficient in offering elements such as user tracking, social interaction, motivation management, and AI-supported personalization within an integrated structure. Especially in intense and competitive educational environments such as university preparation, there is a growing need for integrated platforms that allow students to systematically track their learning processes, analyze their performance, and access content tailored to their individual needs. The inadequacy of traditional educational tools in meeting these requirements necessitates the development of mobile-based smart learning systems.

In this book chapter, an Android-based, artificial intelligence-supported examination and student monitoring system was designed and implemented to support the academic development of students preparing for university entrance exams. The proposed system is built on a modular and scalable software architecture integrating a mobile client layer, cloud-based backend services, and artificial intelligence components. The mobile application was developed using the Flutter framework, while Firebase infrastructure was employed for user authentication, real-time data storage, and synchronization. Within the scope of artificial intelligence integration, an intelligent assistant module based on the Gemini API was incorporated into the system to provide

question–answer interaction, learning support, and motivation-oriented guidance. The developed mobile application provides an integrated structure including topic-based learning content, video-supported instruction, test-solving modules, mock exam tracking, performance analysis mechanisms, progress visualization, and artificial intelligence-based interaction components. The proposed system enables students to monitor their individual learning processes digitally, evaluate their performance through graphical analyses, and receive personalized support. The proposed architecture provides a comprehensive engineering solution for the development of AI-supported student monitoring systems on mobile platforms, bridging software engineering practices with educational technology applications.

RELATED WORKS

M-learning and AI-based education systems have become one of the fastest-growing research topics in the field of educational technologies in recent years. The widespread use of mobile devices makes it possible to conduct learning processes independently of time and place, while AI technologies make these processes more efficient, personalized, and data-driven. This section examines current studies on mobile learning, AI-supported education systems, learning analytics, and interactive learning approaches, and highlights the trends and limitations in the existing literature.

Studies on mobile learning systems show that these technologies have positive effects on student performance and learning motivation. In their study examining mobile learning applications at the higher education level, Rangel-de Lázaro and Duart [1] revealed that mobile platforms increase students' active participation in learning processes and provide significant improvements in learning outcomes. Similarly, Wu and Zeng [16] analyzed the effects of mobile learning systems on user acceptance and learning performance and noted that interactive content and instant feedback mechanisms significantly support the learning process. Likewise, Valencia-Arias et al. [17], in another systematic review conducted in a university context, showed that mobile learning adoption factors vary according to learner perception and intention to use. This study reveals that student attitudes, technological competencies, and the educational context are critical

factors in integrating mobile technologies into learning processes, which is important for understanding different user requirements in the adoption of student tracking systems. While these studies demonstrate that mobile learning has become an important tool in education, they also show that the systems are mostly limited to content delivery and basic interaction features.

AI-assisted learning systems stand out as a key component expanding the capabilities of mobile learning platforms. A systematic literature review by Yaghmour et al. [10] shows that AI-based adaptive learning systems offer personalized learning experiences by analyzing student behavior. These systems create more effective learning environments by adapting content according to students' learning speed, performance, and needs. Mustfa and Ashiq [18] addressed the role of adaptive learning systems in education and revealed that these systems can dynamically optimize learning processes. However, it is noteworthy that a large portion of these studies remain within a theoretical framework, and examples of real-world application development are limited. A study by Gligorea et al. [9] examined the effects of AI-assisted e-learning systems on personalization and showed that these systems play a significant role in improving students' academic achievement. However, the research emphasizes that these systems are generally focused on individual learning and that the social interaction dimension is not adequately addressed. This situation indicates that mobile learning systems, by focusing solely on individual performance, neglect the social and collaborative aspects of learning.

Studies addressing the integration of mobile learning and artificial intelligence reveal that the combined use of these two technologies significantly enhances the learning experience. Guo et al. [19] investigated the effects of AI-powered mobile learning platforms on students' critical thinking skills and learning engagement, showing that these systems provided higher engagement compared to traditional learning methods. Lakulu et al. [20] classified the AI techniques used in mobile learning environments and noted that machine learning, deep learning, and chatbot systems are widely used in educational applications. However, most of these studies focus on a specific AI technique, and integrated system designs appear to be limited.

Other studies in the field of educational technologies highlight the importance of learning analytics and data-driven decision support systems. Akour et al. [21] predicted mobile learning use intention using machine

learning methods and revealed that analyzing user behavior plays a critical role in system design. However, this study also generally focuses on the data analysis aspect and offers limited explanations on how these analyses can be directly integrated into user experience. Interactive and game-based learning approaches also hold a significant place in the literature. In their studies examining adaptive game-based learning systems, Chiotaki et al. [22] showed that these systems significantly increased student motivation and participation in learning. These findings demonstrate how critical motivational elements are in the learning process. However, most game-based systems focus on a specific learning scenario and their integration with broad educational platforms remains limited.

Research evaluating the performance of adaptive and intelligent systems shows that student monitoring tools are effective in detecting the likelihood of learners failing at an early stage. For example, the Adaptive Intelligence System (Learning Intelligent System – LIS) used in the study by Guerrero-Roldán et al. [23] was developed to evaluate student performance and identify at-risk learners with early warning systems. In this study, it was observed that LIS improved student performance and the system was perceived positively by both students and teachers in the educational processes. The study by Sajja et al. [24] describes personalized learning systems within the framework of AI-powered intelligent assistants. This system aims to enrich the educational experience by offering customized learning paths based on natural language processing, information response, and student queries. Thus, AI-powered assistant systems can make learning processes more interactive and supportive for both teachers and students.

Although recent studies have comprehensively examined mobile learning platforms, online examination systems, and AI-based educational technologies, the vast majority of existing studies address these components independently. Many studies primarily focus on mobile examination and assessment tools, emphasizing usability and content delivery, while others concentrate on intelligent tutoring systems or learning analytics frameworks without providing a fully functional mobile assessment infrastructure. Furthermore, many proposed AI-powered education models remain at the conceptual or simulation level, lacking comprehensive engineering implementations demonstrating real-time data management, scalable cloud integration, and end-to-end system deployment. In this context, one of the

most significant gaps in the literature is the inadequacy of integrated systems that combine mobile learning, AI, learning analytics, and social interaction components under a single platform. This situation indicates the need for new approaches to make learning processes more effective, sustainable, and user-centric. Therefore, this study aims to address these gaps in the literature by developing an integrated and AI-powered mobile learning platform, thus distinguishing itself from existing studies in this respect.

METHODOLOGY

A modular and scalable architectural approach was adopted in the system design process. The mobile application, backend infrastructure, and artificial intelligence components are structured to work independently but in an integrated manner. Accordingly, modules with intensive user interaction (test solving, topic learning, performance tracking) were developed on the client side, while data management and business logic processes were configured on the server side. Real-time data flow and low latency were prioritized in the system design. Figure 1 shows the general structure of the developed AI-powered mobile application.

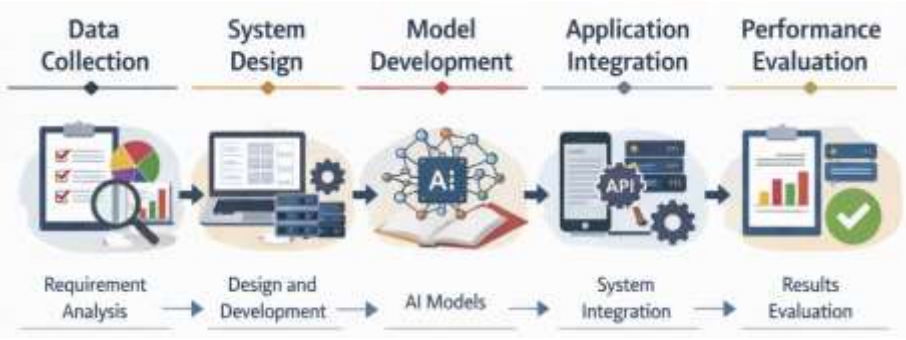


Figure 1. Workflow of the proposed AI-supported mobile learning system

The data collection process is based on behavioral data obtained from users' interactions with the system. This data includes parameters such as completed tests, correct/incorrect rates, subject-based success levels, application usage frequency, and learning time. The collected data was used as the primary input in creating users' learning profiles. This approach allows

for the creation of a dynamic and continuously updated learning structure instead of static learning models.

In the development process of the artificial intelligence component, machine learning and natural language processing techniques were utilized. As a result of analyses performed on user performance data, students' strengths and weaknesses were identified, and appropriate content recommendation mechanisms were developed. In particular, unsupervised learning and classification algorithms were used to group users with similar learning behaviors, and recommendations specific to these groups were provided. In addition, interactive learning was supported by providing answers to users' immediate questions using a natural language processing-based chatbot system.

In the system integration phase, the interoperability of all components was tested. Data communication between the mobile application and the backend system was optimized, API-based data exchange was provided, and the artificial intelligence modules were integrated into the system. In this process, data security and user privacy were prioritized, and authentication and data encryption mechanisms were implemented.

In the performance evaluation process, both the technical and user-oriented performance of the system were analyzed. The technical evaluation included examining the system's response time, data processing speed, and scalability. The user-oriented evaluation considered metrics such as changes in students' academic performance, frequency of system use, and user satisfaction. Furthermore, the accuracy and effectiveness of the system's personalized recommendation mechanism were measured, and the performance of the recommendation algorithms was analyzed.

Finally, based on all the findings, the overall effectiveness of the system was evaluated, and it was shown that the proposed approach offers a more integrated, adaptive, and user-oriented structure compared to traditional mobile learning systems. This methodological framework ensures that the study has a strong foundation in both academic and practical terms.

SYSTEM ARCHITECTURE

The mobile learning system proposed in this study is designed with a multi-layered architectural approach to offer a user-centric, scalable, and AI-supported structure. The system architecture is built on an integrated structure consisting of a mobile client, a cloud-based server infrastructure, and AI components. Thanks to this multi-layered structure, the system can meet both high performance and flexibility requirements; it also enables the efficient processing of user data and the production of meaningful outputs. The proposed system architecture consists of three main components: a client layer, a server layer, and an AI layer. These layers work in continuous data exchange with each other, creating a dynamic and adaptive learning environment. Every interaction performed by the user is processed and analyzed by the other components of the system and transformed into a feedback mechanism. In this context, the system is designed not only as a structure that stores data, but also as a continuously learning and self-updating ecosystem. Figure 2 presents the layered architectural structure of the developed mobile application.

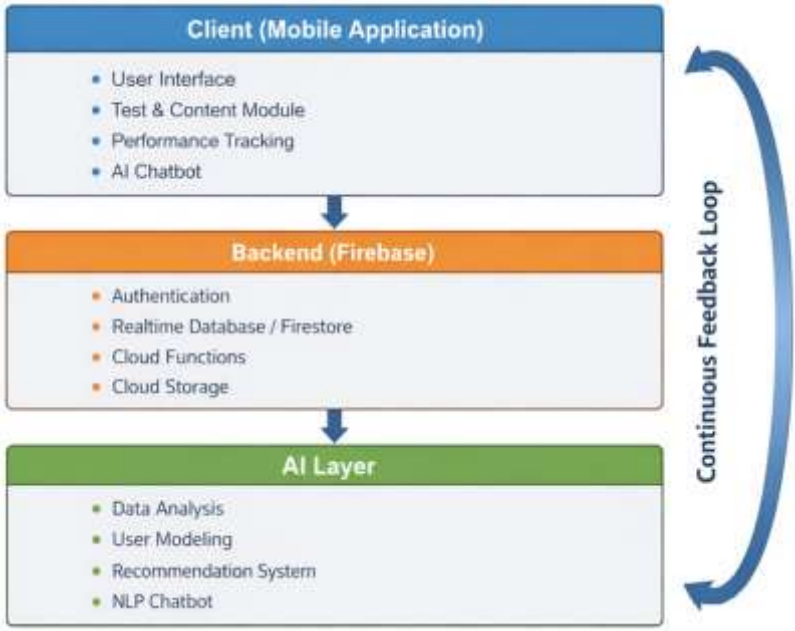


Figure 2. Proposed multi-layered system architecture

The client layer represents the mobile application interface where users interact directly with the system. This layer is implemented using Flutter technology, which offers platform-independent development capabilities, and

features an interface design that prioritizes user experience. The mobile application incorporates various modules that allow users to actively manage their learning processes. These modules include user authentication, subject-based learning content, test-solving mechanisms, performance monitoring tools, an AI-powered chat system, and social interaction components. All interaction data obtained from the user is instantly transmitted to the backend system, enabling real-time analysis and feedback processes.

The server layer constitutes the central component where the system's data management, business logic, and security processes are executed. In this study, the cloud-based Firebase platform was chosen as the backend infrastructure. Thanks to Firebase's scalability and high availability features, the system can adapt to different user densities. Within this layer, user authentication processes are performed securely, user data is stored in a real-time database, and all data flow within the application is controlled. In addition, the system's business logic is managed through cloud functions, and data processing processes are automated. This structure both improves the system's performance and simplifies maintenance and update processes.

The artificial intelligence layer is one of the most critical components of the system and is responsible for analyzing user data and creating personalized learning experiences. In this layer, user behavior is analyzed using machine learning and natural language processing techniques, and dynamic recommendation systems are developed based on these analyses. Individual learning profiles are created by considering users' test performance, learning habits, and interaction levels, and content suitable for these profiles is presented. In addition, thanks to the natural language processing-based chatbot system, users' immediate questions are answered, and the learning process is made interactive. This approach ensures that learning is considered not only as content consumption but also as an interactive process.

Data flow within the system occurs in a continuous and cyclical manner based on user interactions. Every action performed by the user via the mobile application is transmitted to the backend layer and recorded there. The collected data is then sent to the AI layer for analysis when necessary, and the results are sent back to the client layer. This process forms the basis of the real-time feedback mechanism. Furthermore, the system has a self-updating structure based on continuous data generation and analysis. This allows the user experience to become more personalized and effective over time.

The proposed system architecture offers several advantages to meet the requirements of modern mobile learning systems. Thanks to its cloud-based infrastructure, the system has high scalability and accessibility. Its real-time data processing capacity enables instant feedback to the user, making the learning process more effective. AI-powered recommendation mechanisms aim to improve users' academic performance by providing a personalized learning experience. Additionally, the modular architecture supports future-proofing of the system and allows for easy integration of new components.

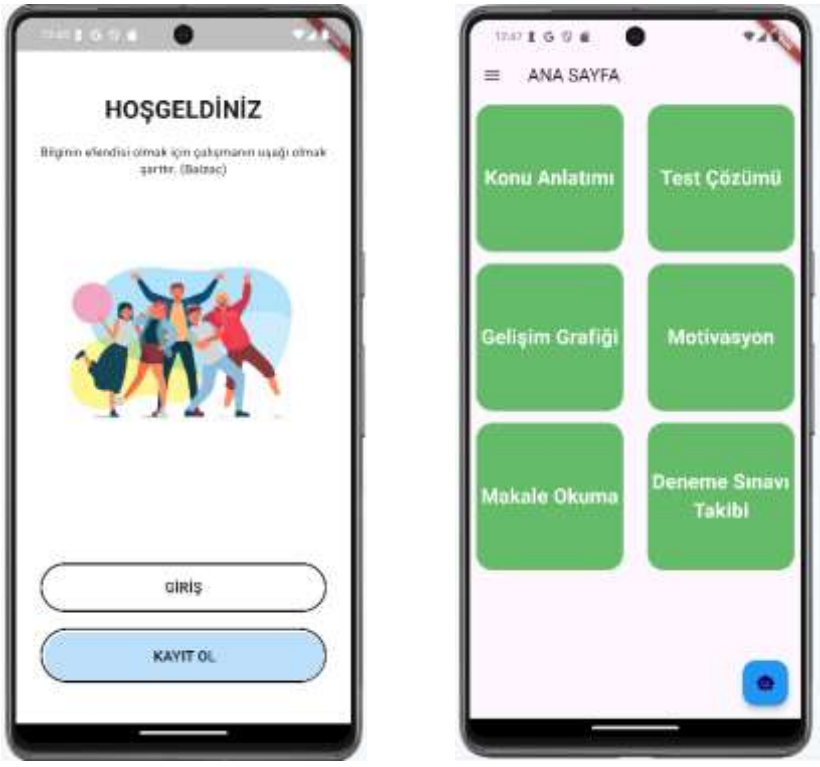
MOBILE USER INTERFACE DESIGN

The success of mobile learning systems depends not only on the technical infrastructure but also directly on the quality of interaction between the user and the system. Therefore, the mobile interface of the proposed system was designed based on user experience (UX) and usability principles. During the design process, increasing user active participation in the learning process, reducing cognitive load, and ensuring intuitive use were among the primary goals. In this context, the interface design was approached within a user-centered design framework and developed considering different user scenarios.

The user authentication process is presented with a simple and effective interface to ensure system security and the protection of user data. The login and registration screens are designed to allow users to quickly access the system with minimal information. To reduce user cognitive load, unnecessary form fields have been avoided, and verification processes are performed automatically in the background. This creates a secure access mechanism without interrupting the user experience. Figure 3.a shows the login/registration page of the developed mobile application.

The overall architecture of the mobile interface is built on a hierarchical structure to allow users to navigate the system quickly and efficiently. A simple and understandable design has been adopted in the navigation structure, aiming to enable users to access basic functions in a minimum number of steps. In this context, transitions between the main page (dashboard), training modules, test screens, performance analysis, and AI-powered chat components have been optimized. The color palette, iconography, and typography elements used in the interface have been

standardized to ensure consistency and improve user experience. Figure 3.b shows the general navigation page of the developed mobile application.



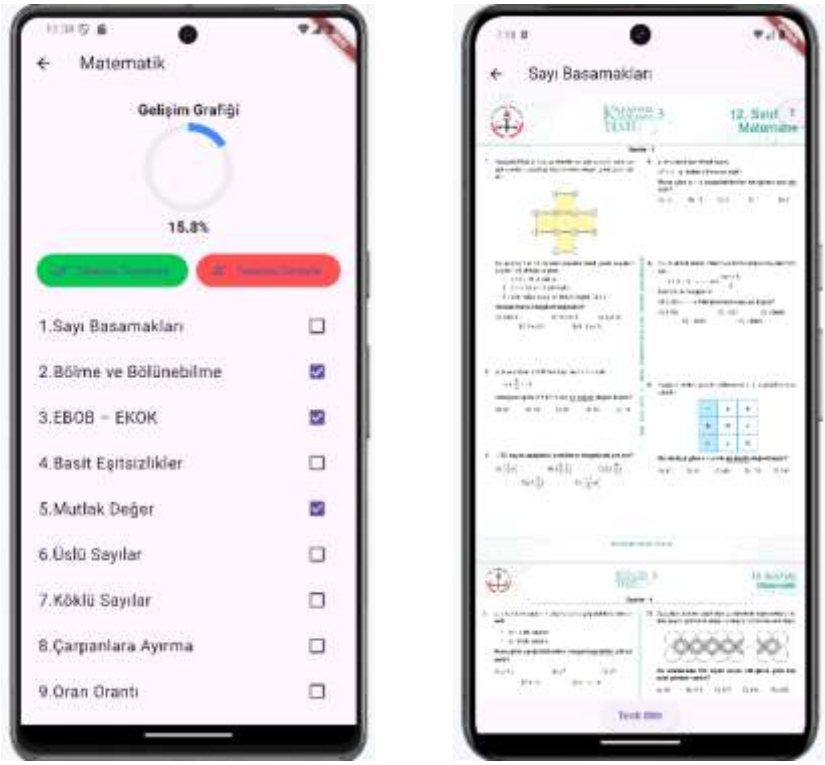
(a) (b)

Figure 3. Login and Dashboard of the mobile application

The main dashboard is the core interface component where users first interact with the system and gain an overview of their learning process. This screen presents users' progress, completed tasks, success rates, and system-recommended content in an integrated manner. Designed according to information hierarchy principles, the dashboard screen highlights the most critical information, ensuring users' attention is directed to important data. Furthermore, the graphical representation of user performance using visual data presentation techniques contributes to making the learning process more understandable. Figure 4.a shows the user-responsive progress, statistics, and development graph page in the developed mobile application.

The education and test examining interfaces are among the most important components where users actively participate in the learning process.

The lesson presentation screens are presented with a simple and non-distracting design, allowing users to focus on the content. The test-solving interface prioritizes fast interaction and instant feedback mechanisms. Feedback on users' correct and incorrect answers is structured to support the learning process. Additionally, question transitions and navigation processes have been optimized to improve user experience. Figure 4.b shows the test-solving interface and question navigation structure in the developed mobile application.

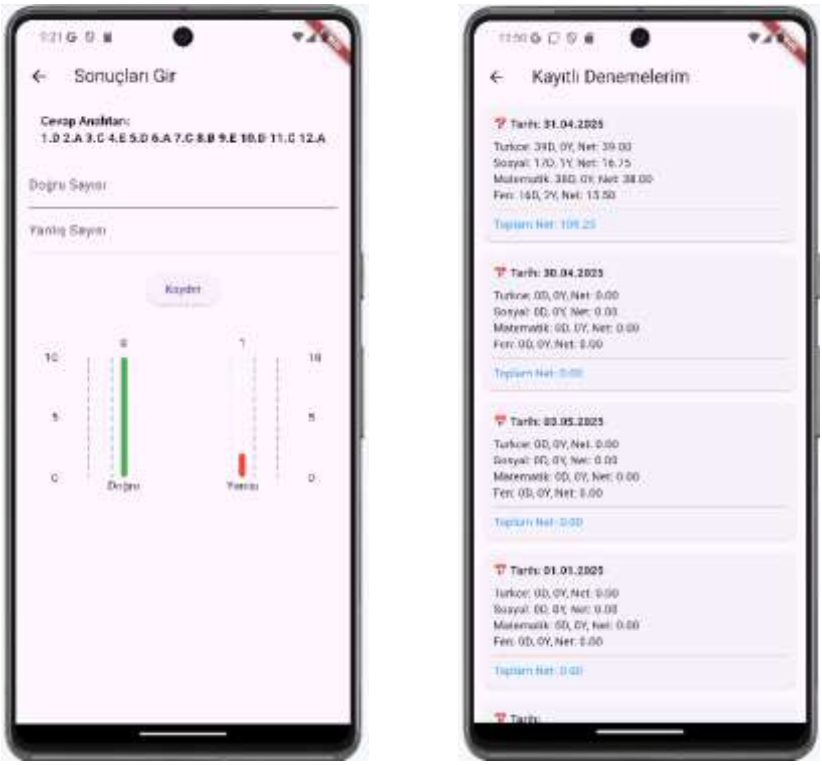


(a) (b)

Figure 4. User progress and Test-solving interfaces

Performance and analysis screens are a crucial component reflecting the system's data-driven approach. These screens present users with data on their learning processes through graphs and statistical indicators. Correct/incorrect answer rates, subject-based success levels, and development trends over time allow users to evaluate their own performance. Such visualization techniques, within the scope of learning analytics, provide meaningful feedback to the

user, enabling more informed management of the learning process. Figure 5.a shows the performance analysis interface, which includes a graphical visualization of learning progress in the developed mobile application, and Figure 5.b shows the interface for recorded practice exam results.



(a) (b)

Figure 5. Performance analysis and exam results interfaces

The AI-powered chat interface stands out as one of the most innovative components of the system. This interface enables users to receive immediate answers to their questions and be supported in the learning process. Developed using natural language processing techniques, this structure creates an interactive communication environment between the user and the system. The AI module is integrated by making RESTful calls to external services within the framework of an API-first architecture. This module provides NLP-based query-response systems and offers text-based feedback to the student. The chat interface has a simple and user-friendly design, offering an experience similar to messaging applications and facilitating user adaptation. Figure 6

shows the AI-powered chatbot interface for interactive learning support in the developed mobile application.

Figure 6. AI-powered chatbot interface

CONCLUSION

This study proposes an integrated system supported by artificial intelligence and natural language processing, aiming to improve user experience and learning efficiency in mobile learning environments. The developed system integrates mobile application infrastructure, cloud-based data management, and artificial intelligence components to analyze data obtained from user interactions and provide personalized learning experiences based on this data. In this respect, the study contributes to transforming mobile learning systems from mere content-providing platforms into user-oriented and adaptive structures.

Experimental results have shown that the proposed system delivers successful results in terms of both technical performance and user experience.

In particular, it has been determined that users participate more actively in the learning process and experience significant improvements in performance thanks to the AI-supported recommendation mechanisms. Furthermore, the natural language processing-based chatbot component supports the learning process and increases the level of interaction by providing quick and effective answers to users' immediate questions. These findings demonstrate that the combined use of mobile applications, artificial intelligence, and natural language processing technologies provides significant advantages in education systems.

Another significant contribution of this study is the integrated presentation of mobile learning, artificial intelligence, and learning analytics components often discussed separately in the literature under a single system. This integrated approach enables end-to-end processing of user data and allows the system to transform into a continuously learning structure. Furthermore, the proposed architecture offers easy adaptability to different application areas thanks to its scalability and modularity.

However, the study also has some limitations. Firstly, the system's performance can vary depending on the size and diversity of the dataset used. Studies conducted with larger and more heterogeneous datasets will increase the generalizability of the system. Furthermore, testing the natural language processing component in different languages and contexts is considered a significant area for improvement towards multilingual and cross-cultural use of the system. Supporting the artificial intelligence models used in the current study with more advanced deep learning approaches is another important factor that can improve system performance.

In conclusion, this study demonstrates the potential of intelligent learning systems developed through the integration of mobile applications, natural language processing, and artificial intelligence technologies, and provides a strong foundation for future research in this field. The proposed approach is applicable not only in education but also in different disciplines such as healthcare, industry, and smart city applications.

REFERENCE

[1] Lazaro, G.Rd., Duart, J.M. Moving Learning: A Systematic Review of Mobile Learning Applications for Online Higher Education. Journal of New

Approaches in Educational Research, 12, 198–224 (2023). https://doi.org/10.7821/naer.2023.7.1287.
[2] Ala, B., & Najah, L. M. (2024). Tutorials and mobile learning in higher education: Enhancing and accessibility. Advances in Mobile Learning Educational Research, 4(1), 920-926. https://doi.org/10.25082/AMLER.2024.01.003.
[3] Naveed, Q. N., Choudhary, H., Ahmad, N., Alqahtani, J., & Qahmash, A. I. (2023). Mobile Learning in Higher Education: A Systematic Literature Review. Sustainability, 15(18), 13566. https://doi.org/10.3390/su151813566.
[4] Alrasheedi, M., Capretz, L. F., & Raza, A. (2015). A Systematic Review of the Critical Factors for Success of Mobile Learning in Higher Education (University Students' Perspective). Journal of Educational Computing Research, 52(2), 257-276, https://doi.org/10.1177/0735633115571928.
[5] Mohammad Abduljawad, A. A. (2023). An Analysis of Mobile Learning (M-Learning) in Education. Multicultural education, 9(2), 145. https://doi.org/10.5281/zenodo.7665894.
[6] Sisouvong, V., & Pasanchay, K. (2024). Mobile Learning: Enhancing Self-Directed Education through Technology, Wireless Networks, and the Internet Anytime, Anywhere. Journal of Education and Learning Reviews, 1(2), 39–50. https://doi.org/10.60027/jelr.2024.752.
[7] J. Garzón, D. Burgos, Kinshuk, A. Tlili, Mobile learning significantly enhances student learning gains: A meta-analysis and research synthesis, Computers & Education, Volume 238, 2025, 105415, https://doi.org/10.1016/j.compedu.2025.105415.
[8] Pedraja-Rejas L, Muñoz-Fritis C, Rodríguez-Ponce E, Laroze D. Mobile Learning and Its Effect on Learning Outcomes and Critical Thinking: A Systematic Review. Applied Sciences. 2024; 14(19):9105. https://doi.org/10.3390/app14199105.
[9] Gligorea, I., Cioca, M., Oancea, R., Gorski, A.-T., Gorski, H., & Tudorache, P. (2023). Adaptive Learning Using Artificial Intelligence in e-Learning: A Literature Review. Education Sciences, 13(12), 1216. https://doi.org/10.3390/educsci13121216.
[10] S. Yaghmour, M. I. Qureshi, I. Ullah, R. K. Perumal, and M. M. Khan, AI-Driven Adaptive Learning Systems in Mobile Education: A Systematic Review of Personalization Strategies, Effectiveness, and User Interaction, International Journal of Interactive Mobile Technologies (iJIM), 19(19), 70–86, 2025. https://doi.org/10.3991/ijim.v19i19.57695.
[11] Y. Özkan, T. Kışla, A Systematic Review of AI-Based Mobile Learning Environments: Unveiling Trends and Future Directions. Journal of Computer Education, 3(1): 1-24, 2024. https://doi.org/10.3991/ijim.v19i19.57695.
[12] R. Alfredo, V. Echeverria, Y. Jin, L. Yan, Z. Swiecki, D. Gašević, R. Martinez-Maldonado, Human-centred learning analytics and AI in education: A systematic literature review, Computers and Education: Artificial Intelligence, Volume 6, 2024, 100215, https://doi.org/10.1016/j.caeai.2024.100215.
[13] Ali, A.A.A., & Dauwed, M. (2022). Development of A Hybrid Mobile App for Student Management System. Journal of Advanced Sciences and Nanotechnology, 1(1), 37–42. https://doi.org/10.55945/joasnt.2022.1.1.37-42.
[14] Rania Lampou, Maria Noor, Raveenthiran Vivekanantharasa, & Sambhabi Patnaik. (2026). AI–Driven Educational Ecosystems: Integrating Learning Analytics, Adaptive Assessment, and Intelligent Feedback for Sustainable

Student Performance. Qubahan Techno Journal, 5(1), 25-36. https://doi.org/10.48161/qtj.v5n1a80.
[15] Sajja R, Sermet Y, Cikmaz M, Cwiertny D, Demir I. Artificial Intelligence-Enabled Intelligent Assistant for Personalized and Adaptive Learning in Higher Education. Information. 2024; 15(10):596. https://doi.org/10.3390/info15100596.
[16] Wu, Q., Ze, X., Application of Mobile Learning Platform in Intelligent Higher Education System, Journal of Electrical Systems, 20-9s(2024):1432-1437, 2024.
[17] Valencia-Arias A, Cardona-Acevedo S, Gómez-Molina S, Vélez Holguín RM, Valencia J. Adoption of mobile learning in the university context: Systematic literature review. PLoS One. 2024 Jun 7;19(6):e0304116. doi: 10.1371/journal.pone.0304116. PMID: 38848384; PMCID: PMC11161115.
[18] Muhammad Jawad Mustfa, Sidra Ashiq, "Harnessing the Power of Artificial Intelligence for Adaptive Learning Systems: A Systematic Review", International Journal of Education and Management Engineering (IJEME), Vol.14, No.5, pp. 12-22, 2024. https://doi.org/10.5815/ijeme.2024.05.02.
[19] Guo, S., Halim, H.B.A. & Saad, M.R.B.M. Leveraging AI-enabled mobile learning platforms to enhance the effectiveness of English teaching in universities. Sci Rep 15, 15873 (2025). https://doi.org/10.1038/s41598-025-00801-0.
[20] Muhammad Modi Lakulu, Ayad Shihan Izkair, Mohd Fadhil Harfiez Abdul Muttalib, Nur Azlan Zainuddin, Leveraging AI in mobile learning to support education: A taxonomy of AI applications, Journal of Infrastructure, Policy and Development, Vol 8, Issue 16, 7347, 2024. https://doi.org/10.24294/jipd7347.
[21] Iman Akour, Muhammad Alshurideh, Barween Al Kurdi, Amel Al Ali, Said Salloum, Using Machine Learning Algorithms to Predict People's Intention to Use Mobile Learning Platforms During the COVID-19 Pandemic: Machine Learning Approach, JMIR Medical Education, Volume 7, Issue 1, 2021, https://doi.org/10.2196/24032.
[22] Chiotaki D, Poulopoulos V and Karpouzis K (2023) Adaptive game-based learning in education: a systematic review. Front. Comput. Sci. 5:1062350. doi: 10.3389/fcomp.2023.1062350.
[23] Guerrero-Roldán, AE., Rodríguez-González, M.E., Bañeres, D. et al. Experiences in the use of an adaptive intelligent system to enhance online learners' performance: a case study in Economics and Business courses. Int J Educ Technol High Educ 18, 36 (2021). https://doi.org/10.1186/s41239-021-00271-0.
[24] Sajja, R., Sermet, Y., Cikmaz, M., Cwiertny, D., & Demir, I. (2024). Artificial Intelligence-Enabled Intelligent Assistant for Personalized and Adaptive Learning in Higher Education. Information, 15(10), 596. https://doi.org/10.3390/info15100596.

A Systematic Comparison of Convolutional and Vision Transformer Architectures on the TN5000 Dataset

Esra YÜCE[1]
Esra GÜNGÖR ULUTAŞ[2]
Mücella ÖZBAY KARAKUŞ[3]

1- Res. Asst.; Yozgat Bozok University, Faculty of Engineering-Architecture, Department of Computer Engineering. esra.yuce@yobu.edu.tr ORCID No: 0000-0002-9522-8352
2- Lect.; Yozgat Bozok University, Sorgun Vocational School, Department of Computer Technologies. esra.ulutas@bozok.edu.tr ORCID No: 0000-0001-8084-6644
3- Assoc. Prof. Dr.; İzmir Bakırçay University, Faculty of Engineering and Architecture, Department of Computer Engineering. mucella.ozbaykarakus@bakircay.edu.tr ORCID No: 0000-0003-0599-8802

ABSTRACT

Thyroid nodule malignancy classification from ultrasound images remains a challenging clinical problem due to subtle inter-class visual differences and significant class imbalance in biopsy-referred populations. In this study, two fundamentally distinct deep learning architectures — ConvNeXt-Small, representing the convolutional approach, and DINOv2 ViT-B/14, representing the self-supervised Vision Transformer approach — are systematically compared on TN5000, one of the largest publicly available thyroid ultrasound datasets comprising 5,000 biopsy-confirmed images. A region-of-interest (ROI)-centered input representation with 1.4× contextual padding was adopted to capture both nodule internals and surrounding parenchymal tissue within a controlled field of view. To address class imbalance, focal loss, weighted random sampling, and Mixup augmentation were combined within a two-phase training protocol. Model evaluation was performed using 5-fold stratified cross-validation on a fixed training pool, with all final results reported on a held-out test set unseen during training or validation. Test-time augmentation with 8-fold averaging was applied at inference. ConvNeXt-Small achieved an ensemble AUROC of 0.955 and sensitivity of 0.969, while DINOv2 ViT-B/14 attained an ensemble accuracy of 0.919 and F1-score of 0.944, with both models surpassing all previously published results on TN5000. Grad-CAM++ visualizations confirmed that both architectures attend to clinically meaningful nodule regions, with complementary attention patterns consistent with their respective inductive biases. These findings demonstrate that ROI-centered representation and robust training strategy contribute substantially to classification performance, and that CNN and Vision Transformer architectures offer complementary strengths suited to different clinical priorities.

Keywords – thyroid nodule classification, ultrasound imaging, deep learning, convolutional neural network, Vision Transformer

INTRODUCTION

Thyroid carcinoma is the most frequent endocrine tumor, which has been showing an increasing incidence trend over the last decades. According to GLOBOCAN 2022 estimates from the International Agency for Research on Cancer (IARC), an estimated 821,214 new thyroid cancer cases occurred worldwide in 2022, making it the most common endocrine malignancy (Lyu et al., 2024). This growing incidence rate could be partially attributed to improved imaging techniques, in particular ultrasonography (US), but not only (Lim et al., 2017).

Thyroid nodules are highly common among the population, with prevalence ranging between 19–67% in adults undergoing high-resolution ultrasonography examination (Guth et al., 2009). Yet only 7–15% of the lesions turn out to be malignant in nature (Haugen et al., 2016). Malignant nodule detection is undoubtedly the most important diagnostic problem faced by clinicians working in thyroid diseases management since excessive biopsies imply unnecessary surgeries while failing to recognize malignancy may have catastrophic consequences for patients. The application of deep learning in medical image analysis has resulted in remarkable achievements in recent years, giving rise to methods that can successfully compete with professional physicians in dermatology, radiology, and pathology tasks (Esteva et al., 2017; Rajpurkar et al., 2018). In terms of thyroid ultrasonography, convolutional neural networks (CNN) have repeatedly proven their excellent ability to discriminate between benign and malignant nodules (Ko et al., 2019). Recently, Vision Transformer (ViT) architectures (Dosovitskiy et al., 2021) and self-supervised pretraining frameworks (Oquab et al., 2023) have helped solve the inherent limitation of conventional CNN in modeling long-range dependencies, thereby becoming the new gold standard in medical image analysis. Focal loss (Lin et al., 2017) originally introduced for object detection tasks by Lin et al., has found wide application in medical imaging classification due to its modification, which causes harder and misclassified examples to be weighted more heavily. Augmentation technique Mixup (Zhang et al., 2018) creates artificial samples through linear interpolation of two training images and encourages the model to learn the decision boundaries smoothly in transitional regions. Albumentations package (Buslaev et al., 2020) allows for performing effective and convenient data augmentation on medical images and has become increasingly popular in thyroid US studies. Yavuz and Yumuşak (Yavuz & Yumuşak, 2026) used a graph neural network (GNN)-based methodology in thyroid nodule recognition task on the TN5000 dataset. The authors used the 5-fold × 3-seed cross-validation evaluation protocol, obtaining an AUROC score of 0.906 and AUPRC of 0.954. Using a 5-fold × 3-seed protocol for cross-validation is theoretically sound because it reduces sensitivity to random data splitting and provided an idea for the evaluation process of the current work. The ConvNeXt architecture (Liu et al., 2022) updates the classic ResNet framework inspired by Transformers design with such innovations as large-kernel depthwise convolution, layer normalization, and GELU activation function that enables to achieve results comparable or similar to pure ViT architectures. Thus, ConvNeXt becomes the perfect reference architecture for comparing with the Transformer.

The study aims to compare two essential deep learning paradigms for thyroid nodule discrimination on thyroid ultrasonography. Two state-of-the-art models – ConvNeXt-Small (Liu et al., 2022), representative of the CNN

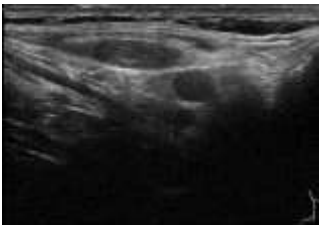
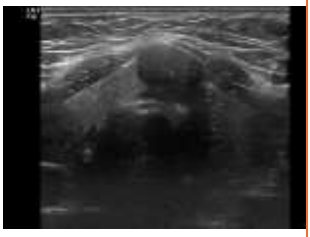
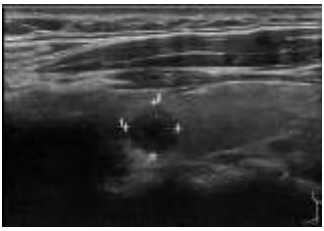
architecture, and DINOv2 ViT-B/14 (Oquab et al., 2023), representative of the self-supervised vision transformers approach will be investigated in this study. Both models will be evaluated on TN5000 (Zhang et al., 2025) dataset, which is currently one of the most extensive thyroid US datasets available for researchers, based on a 5-fold cross-validation scheme. Contributions of this study can be summarized as follows: (i) designing a strategy of ROI-based input preparation with context padding; (ii) using combination of focal loss, weighted sampling, and Mixup as measures against class imbalance; (iii) comparing CNN and Vision Transformer architectures under the same evaluation protocol; and (iv) interpreting model decisions with Grad-CAM++.

MATERIALS AND METHODS

Dataset

The dataset used in this experiment was the TN5000 (Zhang et al., 2025), which is among the largest open-source repositories of thyroid ultrasound images. The dataset includes 5,000 thyroid ultrasound images collected from the National Cancer Center of China using high-frequency linear transducers with frequencies ranging between 5-15 MHz. The images are annotated with a corresponding pathology diagnosis based on fine-needle aspiration biopsy (FNA) cytology or surgical histopathology. The database consists of 1,426 benign lesions (28.5%) and 3,574 malignant lesions (71.5%) (Haugen et al., 2016).

Table 1. Class distribution and sample ultrasound images from the TN5000 dataset.

Classes	**Number of samples**	**Sample Images**	
Benign	1426		
Malignant	3574		

All images are annotated in PASCAL VOC format with bounding boxes delineating nodule boundaries. The annotation process was carried out in three stages: initial labeling by junior radiologists, verification by a senior radiologist with over 30 years of experience, and final confirmation through biopsy and pathological diagnosis. The majority of images are standardized to a resolution of approximately 718 × 500 pixels.

Data Pre-Processing

In this study, a region-of-interest (ROI)-centered input representation was adopted rather than using the full image. In clinical practice, radiologists evaluating thyroid nodules examine not only the interior of the nodule but also the surrounding tissue characteristics, including the halo sign, margin irregularity, and posterior acoustic features (Tessler et al., 2017). While full-image approaches include diagnostically irrelevant regions such as neck soft tissue and device artifacts, isolated ROI cropping discards valuable peri-lesional information. To overcome the limitations of both approaches, a square crop was extracted by centering on the bounding box annotation and expanding 40% in each direction (1.4× padding). This approach captures the nodule, its margins, and the surrounding parenchymal tissue within a single crop under a controlled field of view.

A comprehensive augmentation pipeline was applied to the training set, including horizontal/vertical flipping, shift-scale-rotation, random brightness-contrast adjustment, CLAHE, elastic/grid/optical distortion, and coarse dropout. Mixup augmentation (α=0.4) was additionally applied to generate virtual training samples (Zhang et al., 2018). The validation set was processed with resizing and normalization only. For the test set, 8-fold test-time augmentation (TTA) was applied at inference: the original image, horizontal flip, vertical flip, two mild rotations (±8–12°), brightness-contrast adjustment, center-cropped zoom (1.12×), and Gaussian blur. The final prediction for each test sample was obtained by averaging the sigmoid probabilities across the eight views. All image processing operations were performed using the Albumentations library and normalized with ImageNet statistics. Rather than using the dataset's original split, a custom partitioning strategy was implemented to ensure a more reliable evaluation. The entire dataset was stratified with a fixed random seed into an 80% training pool (4,000 images) and a 20% held-out test set (1,000 images), maintaining the benign/malignant ratio consistent with the original distribution in both subsets. Five-fold stratified cross-validation was performed on the training pool, with 3,200 images used for training and 800 for validation in each fold. The test set remained fixed throughout all experiments and was used exclusively at the final evaluation stage, enabling direct comparison across

architectures while eliminating the risk of data leakage. By preferring cross-validation over a single split, sensitivity of results to any particular random partition was avoided and the consistency of model performance across different data subsets was measured.

Table 2. Sample counts and experimental data partitioning distribution of the TN5000 dataset.

Class Distribution			
	Benign	**Malignant**	**Total**
The entire dataset	1426	3574	5000
Experimental Section			
Training pool (80%)	1141	2859	4000
└ Train (on every fold)	913	2287	3200
└ Validation (on every fold)	228	572	800
Fixed test set (20%)	285	715	1000

Method

In this study, two completely different deep learning architectures were trained and tested in order to perform classification of thyroid nodules. The first architecture belongs to convolutional neural networks (CNN), known for its capabilities to extract spatial hierarchies and local texture information; the second one uses a vision transformer (ViT), which can take into account long-range dependencies over the whole image by means of global self-attention. Both models used the same input representation in the form of a centered ROI crop, the same augmentation pipeline, and the same training procedure in order to compare results directly.

Architecture I – ConvNeXt-Small (CNN): ConvNeXt-Small architecture pre-trained on ImageNet-22K was used as a backbone. ConvNeXt is a CNN architecture that modernizes the old ResNet structure in terms of using large kernels (7x7 depthwise convolutions), inverted bottleneck blocks, layer normalization, and GELU activation. This CNN model reaches Transformer-level accuracy but requires significantly less computation due to fewer fully connected layers. The model accepts a single cropped and padded input image

with size 448x448 pixels and outputs a 768D vector using global average pooling.

Architecture II – DINOv2 ViT-B/14 (Vision Transformer): DINOv2 ViT-B/14 is a vision transformer that learns self-supervised representations via self-distillation from 142 million images. Representations learned on unlabeled data show especially good transfer capabilities in cases when there is a significant domain shift between natural and medical images. The ViT model works with a single cropped and padded input image with the size 518x518 pixels. For fine-tuning purposes, layer-wise learning rate decay (decay rate: 0.65) was applied, which assigns smaller learning rates to the bottom transformer blocks to preserve general properties and larger learning rates to the top transformer blocks for task-specific learning. Training was performed in two stages. In the first stage (5 epochs), all the backbone weights were frozen, and the training took place solely on a classification head to obtain a fast and clean initialization. In the second stage (40-50 epochs), the whole model was fine-tuned with a differential learning rate strategy (backbone - small, head - larger). The learning rate restarts were introduced via CosineAnnealingWarmRestarts scheduler. To handle class imbalance, three techniques were applied: weighted sampling for balanced mini-batches, focal loss function (focal gamma = 2.0), and label smoothing (label smooth value = 0.05). Early stopping with a patience of 15 epochs based on AUROC validation score was used.

Evaluation Metrics

Two kinds of outcomes were measured for every architectural configuration. Per-fold mean and standard deviation indicate average model performance along with variance associated with the highest AUROC scores obtained among the best-performing weights in each fold when applied to a certain fixed test set. In order to obtain an ensemble, probability predictions for each sample were averaged from the best-performing weights obtained in each fold, with five folds in total. Since those weights were trained on different train/val splits that had different decision boundaries, diversity allows making errors compensate each other, resulting in improved outcomes. Two metrics based on thresholds were used as the main criteria of model performance. AUROC serves as a criterion of the model's ability to discriminate at any possible threshold. AUPRC is focused on the model performance at the level of the minority class (Saito & Rehmsmeier, 2015). Three additional metrics based on thresholding the prediction were considered. Accuracy, F1 score, and sensitivity were used for further comparisons. Model explainability was analyzed through Grad-CAM++ visualization of predictions made by the models (Chattopadhay et al., 2018). All experiments were performed using

PyTorch 2.x library and a single NVIDIA GeForce RTX 4090 graphics card (24 GB VRAM).

RESULTS

This section presents the classification performance of two distinct architectures on the TN5000 dataset. The training dynamics and individual results of each architecture are first examined, followed by a comparative analysis across architectures and a comparison with the existing literature. Finally, the decision-making processes of the models are interpreted through Grad-CAM++ visualizations.

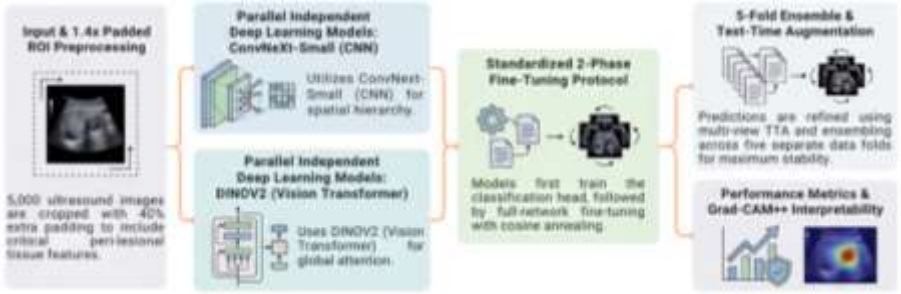


Figure 1. Overview of the proposed method's general workflow.

Figure 2 and Figure 3 show the training curves, ROC curves, PR curves, and confusion matrix of one typical fold from each model architecture. Both models have identical convergence trends that coincide with the two-stage training strategy: fast improvement in the first stage (frozen backbone) and gradual improvements in the second stage (fine-tuning). The validation loss graphs indicate no signs of overfitting, which validates the efficacy of the adopted regularization methods.

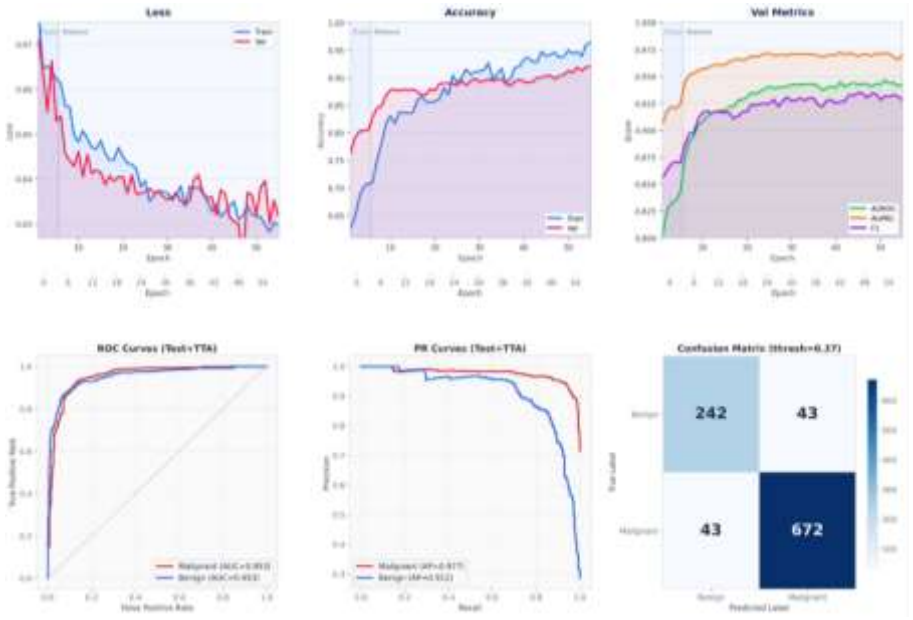


Figure 2. ConvNeXt-Small training results. Top row: loss, accuracy, and validation metric curves. The dashed line indicates the transition from the frozen backbone (Phase 1) to fine-tuning (Phase 2). Bottom row: class-wise ROC curves, PR curves, and confusion matrix.

In the case of ConvNeXt-Small, the loss for the training process followed a consistent reduction pattern; at the same time, the loss for the validation process stabilized after epoch 20. The validation AUROC was

greater than 0.94 throughout, while the AUPRC was greater than 0.96. The ROC curves show equal AUC value of the two classes.

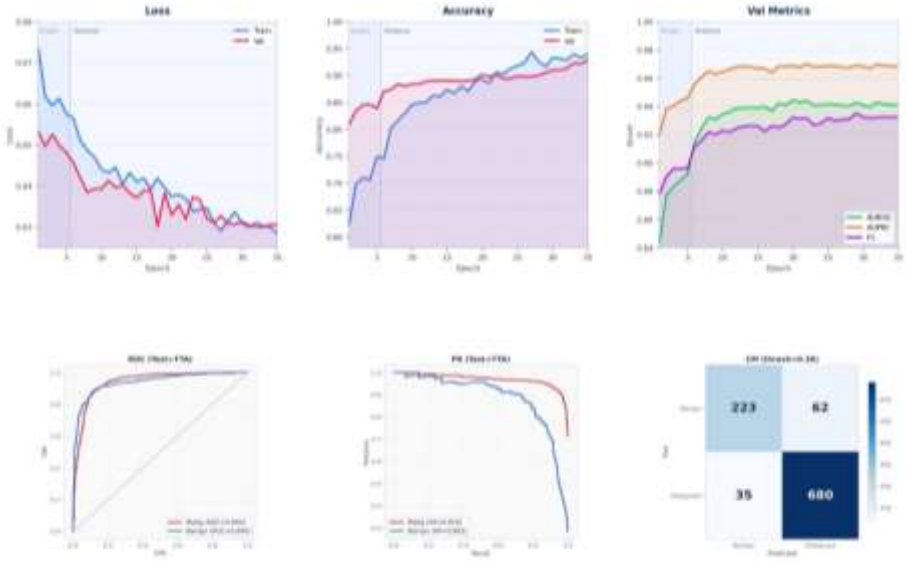


Figure 3. DINOv2 ViT-B/14 training results. Top row: loss, accuracy, and validation metric curves. The dashed line indicates the transition from the frozen backbone (Phase 1) to fine-tuning (Phase 2). Bottom row: class-wise ROC curves, PR curves, and confusion matrix.

DINOv2 (Figure 3) followed a different training trajectory. During the first stage where the backbone remained frozen, the initial performance was significantly better for DINOv2 compared to the CNN model (validation AUROC ≈ 0.89).

Table 3 shows the average performance and ensemble performance for each architecture with the use of TTA in the fixed test dataset. The small standard deviations for each architecture show that the performance is consistently achieved regardless of partitioning.

Table 3. Classification performance with TTA on the fixed test set. Per-fold values represent the mean ± standard deviation of independent evaluations of each fold's best model weights. Ensemble values were obtained by sample-wise averaging of predictions from the five-fold models.

Architecture	Type	Accuracy	AUROC	AUPRC	F1-Score	Sensitivity
ConvNeXt-Small	Per-fold	0.905±0.006	0.950±0.003	0.974±0.002	0.936±0.003	0.961±0.014
	Ensemble	**0.908**	**0.955**	**0.976**	**0.938**	**0.969**
DINOv2 ViT-B/14	Per-fold	0.908±0.005	0.944±0.003	0.972±0.001	0.937±0.004	0.957±0.009
	Ensemble	**0.919**	0.949	0.974	**0.944**	0.954

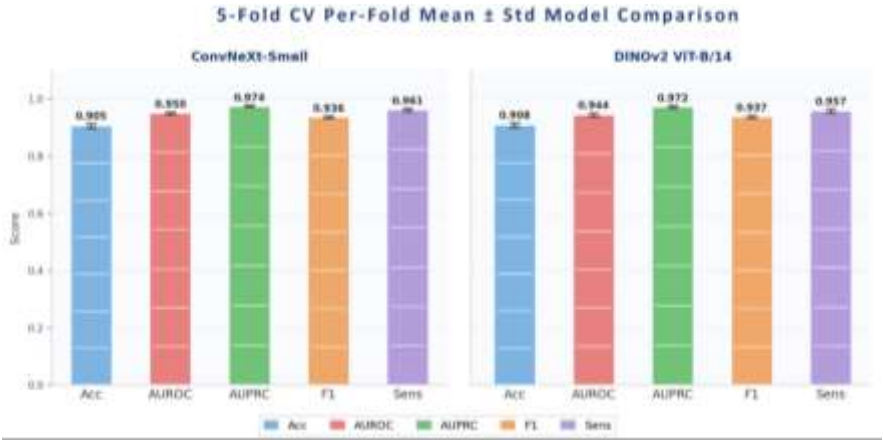


Figure 4. The per-fold mean ± standard deviation of each architecture on primary evaluation measures. The error bar represents one standard deviation between five folds.

According to Table 3 and Figure 4, there are some important insights. Firstly, both models have AUROC between 0.944 and 0.950 and AUPRC between 0.972 and 0.974, showing that the ROI-based method can deliver good performance regardless of the backbone model used. Secondly, ensembling always leads to better results compared with the per-fold mean performance. There are differences among the two architectures in terms of the strengths of each of them. The ConvNeXt-Small obtains the best AUROC (0.955) and sensitivity (0.969), making it the best choice for use in screening applications that need higher sensitivity. Conversely, the DINOv2 reaches the maximum accuracy and F1-score.

Table 4. Comparison with published methods on the TN5000 dataset.

Study	Method	Accuracy	AUROC	AUPRC	F1- Score	Sensitivity
(Zhang et al., 2025)	DETR + ResNet-50	—	—	—	0.850	—
(Bahmane et al., 2025)	EfficientNet-B3 + SE/residual, GAN aug.	0.897	—	—	0.889	0.900
(Yavuz & Yumuşak, 2026)	ROI+context GNN	—	0.906	0.954	0.922	—
(Ours)	ConvNeXt-Small	0.908	**0.955**	**0.976**	0.938	**0.969**
(Ours)	DINOv2 ViT-B/14	**0.919**	0.949	0.974	**0.944**	0.954

As can be seen from Table 4, the performance of both suggested architectures surpasses those achieved in previous studies with the TN5000 dataset based on multiple metrics. Compared to the 5-fold × 3-seed cross-validation results of Yavuz and Yumuşak (Yavuz & Yumuşak, 2026), the ConvNeXt-Small ensemble offers gains of 4.9 points in AUROC, 2.2 points in AUPRC, and 1.6 points in F1-score. As compared with Bahmane et al. (Bahmane et al., 2025), the architecture delivering the best outcome (DINOv2) provides enhancements of 2.2 points in accuracy and 5.5 points in F1-score. Figure 5 shows the Grad-CAM++ activation maps for the two architectures in four different classification scenarios. Specifically, the rows illustrate the true positive (TP; row 1), true negative (TN; row 2), false positive (FP; row 3), and false negative (FN; row 4) cases. For each pair of columns, the original ultrasound image crop is shown alongside the Grad-CAM++ map superimposed onto it. Red/yellow areas represent locations of focus for the classifier.

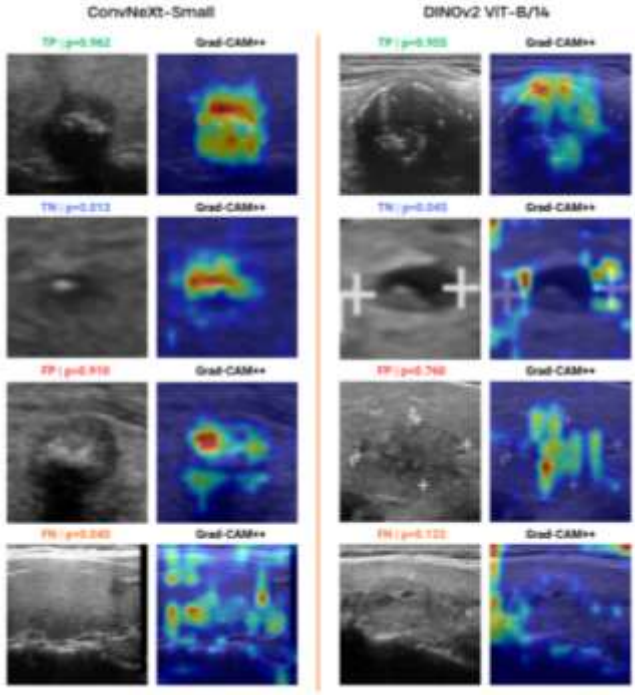


Figure 5. Grad-CAM++ activation maps for the two architectures. Columns from left to right: ConvNeXt-Small and DINOv2 ViT-B/14. Rows from top to bottom: true positive (TP), true negative (TN), false positive (FP), false negative (FN). In each pair, the left side shows the original ROI cropping, and the right side shows the Grad-CAM++ overlay.

In cases where both architectures yield correct predictions (row one), both networks activate consistently inside the nodule in a very local way with high activation on hypoechoic areas and an irregular appearance of the internal structures. ConvNeXt-Small (p=0.962) demonstrates the most localized activation while the activation map obtained by DINOv2 (p=0.955) is less concentrated; however, it reflects the essence of the global attention used by this network. As for true negatives (second row), the models classify them as benign lesions with low probability of being malignant. Attention to the nodule regions was identified in all activations; however, activation was relatively weak and non-localized which indicates that the absence of the malignant criteria causes such a result. False positives are cases when models detect malignancy with high probabilities despite the lesion being benign according to histological analysis. Activation maps demonstrate that visual features resembling the characteristics of the malignant nodules attract the attention of both models which is explained by clinical data. There are certain benign lesions whose imaging parameters resemble those of malignant nodules (Tessler et al., 2017); for instance, follicular adenoma or nodular hyperplasia. In false negative cases (last row), lesions were classified as

benign despite being histologically proven as malignant nodules. It can be observed from activation maps that models pay little attention to the lesions, which means that they lack the visual features associated with malignancy which makes them unrecognizable by automated models. These cases may involve well-differentiated or early stage malignancies.

DISCUSSION

This paper highlights that both ConvNeXt-Small and DINOv2 ViT-B/14 successfully outperform previously presented benchmarks in their classification capabilities with regards to the TN5000 dataset. As hypothesized, both methods leverage the input representation with the contextual information about the lesion and its surrounding tissues successfully, demonstrating comparable results regardless of the utilized architecture. Therefore, ROI-based context appears to be a valuable and meaningful inductive bias in the domain of classification problems for thyroid nodules. In terms of performance metrics, the two models demonstrated complementary strengths: while ConvNeXt-Small provided higher AUROC and sensitivity scores (0.955 and 0.969 respectively), making it better suited for screening applications with higher costs related to missed malignancies, DINOv2 showed slightly higher accuracy (0.919) and F1 score (0.944). These disparities could be explained by the difference in architecture and approach to the problem at hand. Specifically, DINOv2 leverages global self-attention to build representations, while ConvNeXt focuses on local features extracted by CNNs. The use of cross-validation proved beneficial for both architectures: ensemble models consistently achieved higher AUROC and F1-score compared to fold averages, thus serving as validation not only for model evaluation purposes but also as a viable method of building ensembles. Additionally, small standard deviation across different partitions suggests that both models are robust with respect to data distribution in the folds. Grad-CAM++ analysis shows that both models focus on clinically relevant areas during correct predictions. However, false negatives and positives point to one of the key limitations of this project: ambiguity and uncertainty caused by visual similarity between early-stage and benign nodules.

CONCLUSION

In this study, a thorough comparative analysis between CNN and ViT approaches is performed for the thyroid nodules classification task using the TN5000 dataset. Utilizing the robustly developed benchmarking framework, which includes ROI-based input representations, multiple class imbalance mitigation techniques, 5-fold cross-validation with a fixed test set, and test-time data augmentation, both approaches produced highly competitive outcomes. Specifically, the ConvNeXt-Small ensemble demonstrated an

AUROC of 0.955 and a sensitivity score of 0.969, whereas the DINOv2 ViT-B/14 model reached accuracy and F1-score values of 0.919 and 0.944, respectively, surpassing the best results achieved previously for the TN5000 benchmark. It can be noted that the influence of the input representation and training pipeline may be as important as the architecture selection when designing models for clinical tasks, as CNN and ViT models exhibit different advantages. Future research should focus on multi-scale feature learning, attention-based cropping, and clinical validation of the proposed solutions.

REFERENCES

Bahmane, K., Bhattacharya, S., & Chaouki, A. B. (2025). Evaluation of a Hybrid CNN Model for Automatic Detection of Malignant and Benign Lesions. *Medicina, Vol. 61, Page 2036*, *61*(11), 2036. https://doi.org/10.3390/MEDICINA61112036

Buslaev, A., Iglovikov, V. I., Khvedchenya, E., Parinov, A., Druzhinin, M., & Kalinin, A. A. (2020). Albumentations: Fast and Flexible Image Augmentations. *Information, Vol. 11, Page 125*, *11*(2), 125. https://doi.org/10.3390/INFO11020125

Chattopadhay, A., Sarkar, A., Howlader, P., & Balasubramanian, V. N. (2018). Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks. *Proceedings - 2018 IEEE Winter Conference on Applications of Computer Vision, WACV 2018*, *2018-January*, 839–847. https://doi.org/10.1109/WACV.2018.00097

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2021). An image is worth 16×16 words: Transformers for image recognition at scale. *International Conference on Learning Representations (ICLR 2021)*. arXiv:2010.11929.

Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature, 542:7639*, *542*(7639), 115–118. https://doi.org/10.1038/nature21056

Guth, S., Theune, U., Aberle, J., Galach, A., & Bamberger, C. M. (2009). Very high prevalence of thyroid nodules detected by high frequency (13 MHz) ultrasound examination. *European Journal of Clinical Investigation*, *39*(8), 699–706. https://doi.org/10.1111/J.1365-2362.2009.02162.X

Haugen, B. R., Alexander, E. K., Bible, K. C., Doherty, G. M., Mandel, S. J., Nikiforov, Y. E., Pacini, F., Randolph, G. W., Sawka, A. M., Schlumberger, M., Schuff, K. G., Sherman, S. I., Sosa, J. A., Steward, D. L., Tuttle, R. M., & Wartofsky, L. (2016). 2015 American Thyroid Association Management Guidelines for Adult Patients with Thyroid Nodules and Differentiated Thyroid Cancer: The American Thyroid Association Guidelines Task Force on Thyroid Nodules and Differentiated Thyroid Cancer. *Thyroid*, *26*(1), 1–133. https://doi.org/10.1089/THY.2015.0020

Ko, S. Y., Lee, J. H., Yoon, J. H., Na, H., Hong, E., Han, K., Jung, I., Kim, E. K., Moon, H. J., Park, V. Y., Lee, E., & Kwak, J. Y. (2019). Deep convolutional

neural network for the diagnosis of thyroid nodules on ultrasound. *Head and Neck*, *41*(4), 885–891. https://doi.org/10.1002/HED.25415

Lim, H., Devesa, S. S., Sosa, J. A., Check, D., & Kitahara, C. M. (2017). Trends in Thyroid Cancer Incidence and Mortality in the United States, 1974-2013. *JAMA*, *317*(13), 1338–1348. https://doi.org/10.1001/JAMA.2017.2719

Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal Loss for Dense Object Detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 2980–2988). https://doi.org/10.1109/ICCV.2017.324

Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., & Xie, S. (2022). A ConvNet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022)* (pp. 11976–11986). https://doi.org/10.1109/CVPR52688.2022.01167

Lyu, Z., Zhang, Y., Sheng, C., Huang, Y., Zhang, Q., & Chen, K. (2024). Global burden of thyroid cancer in 2022: Incidence and mortality estimates from GLOBOCAN. *Chinese Medical Journal*, *137*(21), 2567–2576. https://doi.org/10.1097/CM9.0000000000003284

Oquab, M., Darcet, T., Moutakanni, T., Vo, H. V., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Assran, M., Ballas, N., Galuba, W., Howes, R., Huang, P. Y., Li, S. W., Misra, I., Rabbat, M., Sharma, V., … Bojanowski, P. (2023). DINOv2: Learning Robust Visual Features without Supervision. *Transactions on Machine Learning Research*, *2024*. https://arxiv.org/pdf/2304.07193

Rajpurkar, P., Irvin, J., Ball, R. L., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Langlotz, C. P., Patel, B. N., Yeom, K. W., Shpanskaya, K., Blankenberg, F. G., Seekins, J., Amrhein, T. J., Mong, D. A., Halabi, S. S., Zucker, E. J., … Lungren, M. P. (2018). Deep learning for chest radiograph diagnosis: A retrospective comparison of the CheXNeXt algorithm to practicing radiologists. *PLOS Medicine*, *15*(11), e1002686. https://doi.org/10.1371/JOURNAL.PMED.1002686

Saito, T., & Rehmsmeier, M. (2015). The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets. *PLOS ONE*, *10*(3), e0118432. https://doi.org/10.1371/JOURNAL.PONE.0118432

Tessler, F. N., Middleton, W. D., Grant, E. G., Hoang, J. K., Berland, L. L., Teefey, S. A., Cronan, J. J., Beland, M. D., Desser, T. S., Frates, M. C., Hammers, L. W., Hamper, U. M., Langer, J. E., Reading, C. C., Scoutt, L. M., & Stavros, A. T. (2017). ACR Thyroid Imaging, Reporting and Data System (TI-RADS): White Paper of the ACR TI-RADS Committee. *Journal of the American College of Radiology*, *14*(5), 587–595. https://doi.org/10.1016/J.JACR.2017.01.046

Yavuz, M., & Yumuşak, N. (2026). ROI+Context Graph Neural Networks for Thyroid Nodule Classification: Baselines, Cross-Validation Protocol, and Reproducibility. *Electronics, Vol. 15, Page 151*, *15*(1), 151. https://doi.org/10.3390/ELECTRONICS15010151

Zhang, H., Cisse, M., Dauphin, Y. N., & Lopez-Paz, D. (2018). mixup: Beyond Empirical Risk Minimization. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*. https://arxiv.org/pdf/1710.09412

Zhang, H., Liu, Q., Han, X., Niu, L., & Sun, W. (2025). TN5000: An Ultrasound Image Dataset for Thyroid Nodule Detection and Classification. *Scientific Data 12:1, 1437*, *12*(1). https://doi.org/10.1038/s41597-025-05757-4

A Comparative Benchmark of Modern Object Detection Models for Afghan License Plate Detection

Esra GÜNGÖR ULUTAŞ[1]
Mücella ÖZBAY KARAKUŞ[2]

1- Lect.; Yozgat Bozok University, Sorgun Vocational School, Department of Computer Technologies. esra.ulutas@bozok.edu.tr ORCID No: 0000-0001-8084-6644
2- Assoc. Prof. Dr.; İzmir Bakırçay University, Faculty of Engineering and Architecture, Department of Computer Engineering. mucella.ozbaykarakus@bakircay.edu.tr ORCID No: 0000-0003-0599-8802

ABSTRACT

This study presents a comparative evaluation of five state-of-the-art object detection architectures—YOLO26m, YOLO11m, YOLOv12m, RT-DETR-L and RF-DETR (Medium)—on the task of Afghan number plate detection. In this study, the original training/validation/test split of the Afghan number plate dataset published on Mendeley Data was preserved; four Ultralytics-based models were trained using the same hyperparameters (image size 640×640, 100 epochs, fixed batch size 32, fixed random seed) via a common API. In RT-DETR-L, automatic mixed precision (AMP) training was disabled due to numerical instability; RF-DETR, however, was trained using its own training API and the recommended hyperparameters. All models were evaluated on a separate test set of 850 images using a common evaluation pipeline (COCOeval) based on pycocotools; accuracy (mAP), F1 at the best working point and inference speed were measured separately. The results show that RF-DETR achieved a statistically significant (paired bootstrap, $p<0.001$) advantage with an mAP@50:95 of 0.682, although this came at the cost of the longest training time (~5.9 hours) and a relatively low inference rate (29.0 FPS). YOLOv12m ranks second, offering a statistically significant advantage; whilst RT-DETR-L, YOLO11m and YOLO26m form a third tier that is statistically indistinguishable from one another. YOLO26m and YOLO11m are recommended for scenarios requiring low latency and limited memory; RF-DETR, on the other hand, is recommended for scenarios where accuracy is a priority.

Keywords – object detection, number plate recognition, YOLO, DETR, comparative evaluation

INTRODUCTION

Vehicle number plate detection is the first and critical step in the automatic license plate recognition (ALPR) process, which forms the basis for applications such as intelligent transport systems, electronic toll collection and traffic enforcement. Over the past decade, deep learning-based object detection architectures have replaced traditional, manually designed feature-based methods (Zhao et al., 2019; Xu et al., 2026); this transition has led to a significant leap in accuracy and robustness across many visual detection tasks, particularly in number plate recognition (Ismail et al., 2025; Laroca et al., 2025). This shift is reflected in the ALPR literature, where detection performance is now reported almost exclusively in terms of deep object detection architectures rather than hand-crafted pipelines.

The primary driving force behind this transformation in the field of object detection has been single-stage detectors. The YOLO (You Only Look Once) architecture, proposed by Redmon et al. (2016), introduced one of the first end-to-end detectors capable of real-time operation by reducing detection to a single regression problem; with its subsequent versions (YOLO11, YOLO26, etc.), it has become the de facto standard within the convolutional neural network (CNN)-based family. In parallel, the DETR (Detection Transformer) architecture, introduced by Carion et al. (2020), reframed object detection as an end-to-end set prediction problem by eliminating manually designed components such as bounding boxes and NMS. RT-DETR (Lv et al., 2023), developed to reduce the high computational cost of DETR, became the first transformer-based detector capable of operating in real time thanks to its hybrid encoder design; YOLOv12 (Tian et al., 2025) incorporated attention mechanisms into the YOLO family at CNN speeds; whilst RF-DETR (Robinson et al., 2025) has stood out in terms of both accuracy and domain adaptability by utilising a pre-trained DINOv2 backbone.

Despite this rapid architectural diversification, there are few studies in which these models have been directly compared under fair conditions on narrow, single-class and region-specific datasets; in particular, no such comparison exists in the literature for a domain with low visual diversity and limited data, such as Afghan number plates. To address this gap, this study systematically compares five architectures—comprising current CNN-based (YOLO26m, YOLO11m, YOLOv12m) and transformer-based (RT-DETR-L, RF-DETR) models—under a common training and evaluation protocol, assessing them across the dimensions of accuracy, speed, model size and training cost.

Faster R-CNN, an older architecture, was deliberately excluded from the benchmark. Instead, five models that are prominent in current detection-accuracy and inference-speed literature were selected three from the CNN-based YOLO family and two from the transformer-based DETR family so that the comparison spans both convolutional and transformer-based current approaches.

MATERIAL AND METHODS

Dataset

In this study, the Roboflow COCO-formatted dataset published on Mendeley Data, associated with the paper titled 'End-to-End Afghan Licence Plate Detection and Recognition Using Deep Learning Models', was utilised (DOI: 10.17632/7vzmbpg296.1). The dataset contains a single object class ("License-Plate"), and the original training/validation/test split has been preserved exactly as it was, without any modifications; thus, all five models were trained and tested on exactly the same images. There are a total of 8,500 images and 9,110 licence plate boxes (Table 1).

The bounding box size distribution was calculated across three categories according to COCO's standard area definition: small ($< 32^2$ pixels), medium (32^2–96^2 pixels) and large ($> 96^2$ pixels). The fact that the proportion of small objects remained consistent at around 9 per cent across the training, validation and test sets indicates that the segmentation is balanced, thereby confirming that small-object performance (AP_small) can be used as a meaningful metric in cross-model comparisons.

Table 1. Image/box counts per dataset split and box-size distribution

Split	Images	Boxes (plates)	Small	Medium	Large
Train	5,950	6,361	8.7%	64.7%	26.6%
Validation	1,700	1,843	9.3%	61.9%	28.8%
Test	850	906	9.2%	59.8%	31.0%
Total	8,500	9,110	—	—	—

During the evaluation phase, an unused placeholder category (id=0) was identified in the ground-truth JSON file for the Roboflow COCO output; bearing in mind that this could disrupt pycocotools' internal category indexing, a 'cleaned' ground-truth file containing only the classes actually used was created, and this file was used in the evaluation (its accuracy was confirmed by feeding the ground-truth values back as predictions, resulting in mAP=1.0).

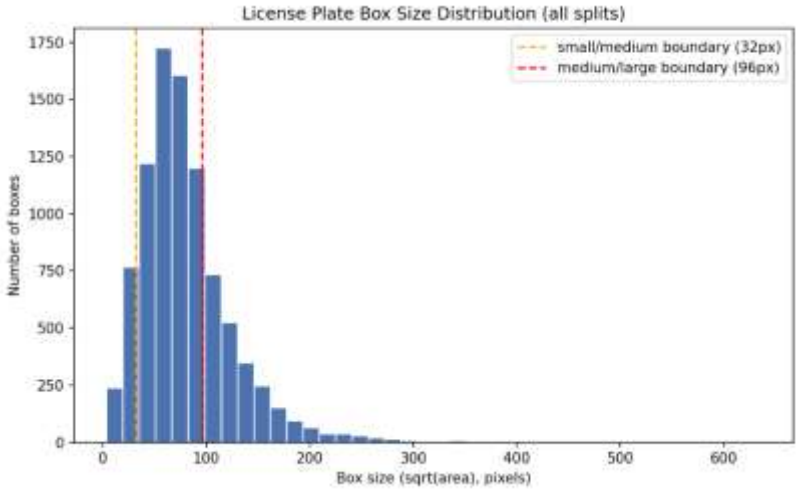


Figure 1: Box-size distribution across splits.

Figure 1 visually compares the distribution of small, medium and large box sizes—as shown numerically in Table 1—for the training, validation and test sets. The graph shows that medium-sized plates (60–65 per cent) are the dominant category in all three sets, with large plates in second place (27–31 per cent) and small plates accounting for the smallest proportion (~9 per cent). The fact that these proportions are similar across the three sections indicates that the split is balanced and, consequently, that size-sensitive results (e.g. AP_small) obtained on the test set will provide a reliable comparison representative of the training distribution.

Models

The five models compared in this study have been selected to represent both CNN- and transformer-based state-of-the-art approaches:

- YOLO26m — The medium-sized variant of Ultralytics' latest YOLO generation; it continues the YOLO family's CNN-based, single-stage detection approach (Redmon et al., 2016).
- YOLO11m — The medium-sized variant of the previous stable YOLO generation; it serves as a widely used benchmark model in terms of the speed-accuracy trade-off.
- YOLOv12m — A medium-sized variant that combines attention mechanisms with a CNN backbone, utilising the representational

power of transformer-based architecture whilst maintaining YOLO's speed (Tian et al., 2025).

- RT-DETR-L — A large variant from the DETR family, featuring a hybrid encoder design to reduce computational cost for real-time operation, and requiring neither bounding boxes nor NMS (Lv et al., 2023).
- RF-DETR (Medium) — A medium-sized variant of a state-of-the-art DETR-based architecture developed by Roboflow, utilising a pre-trained backbone based on DINOv2 (Oquab et al., 2023) and offering high domain adaptability (Robinson et al., 2025).

All models have been adapted to the target dataset using fine-tuning methods based on pre-trained weights (ImageNet/COCO) provided by their respective libraries (Ultralytics and rfdetr). The YOLO11, YOLO26, YOLOv12 and RT-DETR-L implementations were all obtained through the Ultralytics framework (Jocher et al., 2023), which provides a single training and inference API across these architectures; RF-DETR was trained with the rfdetr reference implementation released by its authors.

Evaluation Metrics

The performance of the models was evaluated using metrics considered standard in the object detection literature: mAP@50:95—calculated as the average across the range IoU=0.50:0.95—for overall accuracy, along with mAP@50 and AP@75 at individual thresholds; AP_small/medium/large, broken down by object size; precision, recall and F1 scores at the best operating point (IoU = 0.50); average sensitivity (AR) calculated based on the maximum number of detections; FPS/ms image inference rate, indicating real-time applicability; and the number of parameters, model size, training time and peak VRAM usage, which characterise the model's deployment costs. All accuracy metrics were calculated using the pycocotools-based COCOeval tool via a common, framework-independent code path. These metric definitions follow the COCO evaluation protocol established by Lin et al. (2014).

Training Protocol

To ensure a fair comparison, the four Ultralytics-based models (YOLO26m, YOLO11m, YOLOv12m, RT-DETR-L) were trained through a SINGLE COMMON API, with the following hyperparameters held identical across all four:

Image size: 640×640 (the dataset is already provided at this resolution)

Maximum epochs: 100, with an early-stopping patience of 20 epochs

Batch size: 32 (held FIXED across all models regardless of hardware differences — batch size affects BatchNorm statistics and learning dynamics, making it a prerequisite for a fair comparison)

Random seed: 0

Automatic mixed precision (AMP): enabled — with ONE EXCEPTION: RT-DETR-L. Under AMP, this model repeatedly produced NaN (numerical overflow) losses from the earliest epochs, and even Ultralytics' built-in automatic recovery mechanism (rolling back to the last valid checkpoint and retrying) could not resolve it. This is a numerical-instability issue specific to RT-DETR's transformer-based loss formulation (GIoU + classification + L1), documented in the literature, and is not a “fair comparison” variable in the same sense as batch size, epoch count, or learning rate — it is simply an implementation detail required for training to complete numerically. AMP was therefore disabled for RT-DETR-L only; the other three models were trained with AMP enabled. This exception is flagged with an asterisk (*) in the results tables.

The fifth model, RF-DETR (Medium), uses its own training API, which reads the original COCO-format dataset directly. The effective batch size (16) recommended in its own documentation, independent of GPU capacity, was retained; the learning rate was set to 1e-4, and the epoch count was set to 100 to remain consistent with the other models.

Training was carried out on a workstation with 4× NVIDIA RTX 6000 Pro Blackwell GPUs, with each model locked to its own dedicated GPU via the CUDA_VISIBLE_DEVICES environment variable and run in PARALLEL. This affects only training wall-clock time, not result correctness — each model was trained in its own isolated process, independently of the others.

RESULTS AND DISCUSSION

The performance of the five models was measured on 850 images from a split test set under a common protocol, and the models are presented in order of accuracy (descending mAP@50:95). The table below presents the mAP values for each model—averaged over the range IoU=0.50–0.95along with the AP values at specific thresholds (mAP@50, AP@75) and categorised by object size (small/medium/large).

As can be seen in the table, all models are very close to one another at the mAP@50 level and demonstrate high performance; the main divergence emerges at mAP@50:95, which is a stricter localisation metric, and particularly in small-object performance (where AP is low). These differences are visualised in Figure 2, together with confidence intervals for each model.

Table 2. Accuracy metrics (mAP averaged over IoU=0.50:0.95 and size-conditioned AP). (*) RT-DETR-L was trained with AMP disabled

Model	mAP@50	mAP@50:95	mAP@75	AP small	AP medium	AP large
RF-DETR (Medium)	0.973	0.682	0.800	0.382	0.653	0.780
YOLOv12m	0.974	0.663	0.754	0.384	0.634	0.747
YOLO11m	0.973	0.654	0.732	0.379	0.624	0.742
RT-DETR-L *	0.965	0.654	0.747	0.352	0.628	0.746
YOLO26m	0.966	0.653	0.735	0.365	0.626	0.742

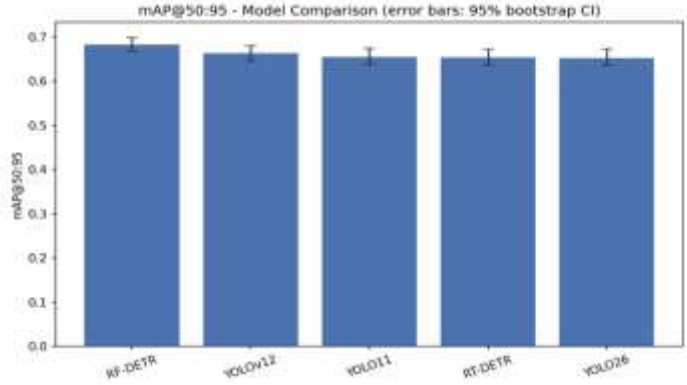


Figure 2: mAP@50:95 — model comparison (error bars: 95% bootstrap confidence interval; see Table 6).

The Table 3 below shows the values at the best F1 point derived from the precision-recall curve. An examination of Figure 3 reveals that, at a practical operating threshold (confidence = 0.25), all models achieve a recall of at least 92 per cent and a precision of over 94 per cent; this indicates that the models perform similarly from an operational perspective.

Table 3. Best F1 point at IoU=0.50 extracted from the precision-recall curve, with the corresponding precision/recall values.

Model	Best precision	Best recall	Best F1
RF-DETR (Medium)	0.980	0.950	0.965
YOLOv12m	0.966	0.950	0.958
YOLO11m	0.967	0.950	0.959
RT-DETR-L *	0.949	0.920	0.934
YOLO26m	0.970	0.950	0.960

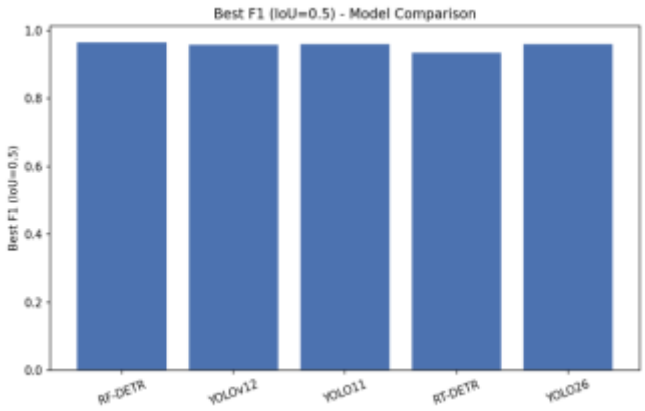


Figure 3: Best F1 (IoU=0.50) — model comparison.

Beyond raw accuracy, practical deployment depends heavily on inference speed, memory footprint, and training cost — axes on which the five architectures diverge sharply. Table 4 reports frames-per-second (FPS) and per-image latency at the confidence=0.25 operating point, alongside parameter count, model file size, total training time, and peak GPU VRAM usage during training. Table 4 shows a clear split between the CNN-based YOLO models and the two transformer-based DETR models. YOLO11m and YOLO26m are the fastest and lightest models in the comparison, reaching roughly 46–47 FPS with the smallest parameter counts and, in YOLO11m's

case, the lowest peak VRAM usage (14.4 GB) — making them the most attractive choices for latency- or memory-constrained deployment. RF-DETR (Medium), despite having a comparable parameter count to RT-DETR-L, is almost exactly twice as large on disk (127.6 MB vs. 63.2 MB) and trains substantially longer than any Ultralytics-based model — about 3.6 to 4.2 times the training time of the YOLO variants — a direct consequence of its separate training pipeline and DINOv2-based backbone. RT-DETR-L, meanwhile, combines the slowest inference speed in the benchmark (27.6 FPS) with the highest peak VRAM consumption (34.5 GB). The elevated memory figure is at least partly attributable to the AMP-disabled training regime discussed earlier, since full-precision activations occupy roughly twice the memory of half-precision ones. The inference speed, by contrast, is a property of the architecture itself and is unrelated to that exception: all five models were timed under identical conditions using their trained weights, and RT-DETR-L in fact completed training in 1.60 hours, marginally faster than YOLOv12m.

Figure 4 visualizes this trade-off directly by plotting FPS against mAP@50:95 for all five models, making the accuracy-speed frontier easy to read at a glance: RF-DETR occupies the high-accuracy, low-speed corner, while YOLO11m and YOLO26m occupy the opposite, high-speed, slightly-lower-accuracy corner, with YOLOv12m and RT-DETR-L positioned in between.

Table 4. Inference speed (confidence threshold=0.25, over 200 images post-warm-up), parameter count, model file size, total training time, and peak GPU VRAM usage during training.

Model	FPS	ms/image	Params (M)	Size (MB)	Training (h)	Peak VRAM (GB)
RF-DETR (Medium)	29.0	34.4	33.4	127.6	5.91	14.5
YOLOv12m	38.3	26.1	20.1	38.9	1.65	20.1
YOLO11m	46.8	21.3	20.1	38.7	1.40	14.4
RT-DETR-L *	27.6	36.2	32.8	63.2	1.60	34.5
YOLO26m	46.1	21.7	21.8	42.0	1.43	16.3

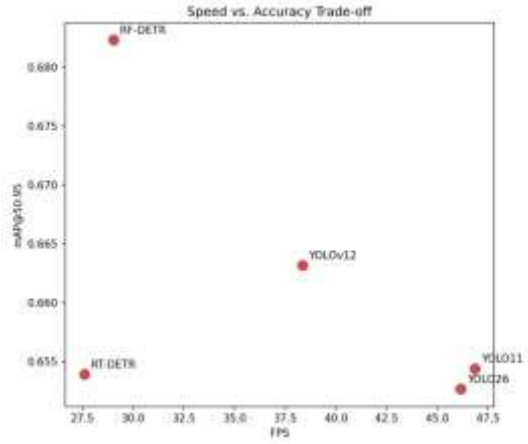


Figure 4: Speed-accuracy trade-off (FPS vs. mAP@50:95).

For completeness, Table 5 reports the Average Recall (AR) values produced by COCOeval, broken down by the maximum number of detections considered (maxDets = 1, 10, 100) and by object size. A notable pattern in Table 5 is that AR@10 and AR@100 are nearly identical for every model. This is not an artifact of the evaluation code, but a direct consequence of the dataset itself: the test set contains only about 1.07 plates per image on average, so the 10-detection cap imposed by AR@10 is, in practice, never binding — no image has enough ground-truth plates to be affected by it. The more informative comparison therefore lies in the size-conditioned columns. Here, RT-DETR-L stands out: although it ranks only mid-table on overall mAP@50:95, it achieves the highest AR on small objects (AR small = 0.655) of all five models, clearly ahead of RF-DETR (0.592) and well ahead of the three YOLO variants (0.543–0.565). This suggests that RT-DETR-L's detection recall on small, difficult plates is stronger than its precision-weighted mAP score alone would indicate — a distinction that Table 2's AP_small figures, being precision-sensitive, do not fully capture. RF-DETR remains the strongest model overall across the medium and large categories, consistent with its lead in Table 2.

Table 5. Average Recall (AR) values, broken down by maximum detections (maxDets) and object size.

Model	AR@1	AR@10	AR@100	AR small	AR medium	AR large
RF-DETR (Medium)	0.702	0.769	0.779	0.592	0.767	0.856
YOLOv12m	0.692	0.726	0.726	0.565	0.715	0.794
YOLO11m	0.690	0.724	0.724	0.559	0.711	0.795
RT-DETR-L *	0.687	0.735	0.748	0.655	0.729	0.812
YOLO26m	0.693	0.725	0.725	0.543	0.715	0.799

Because all five models were evaluated on the identical 850 held-out test images, part of the observed accuracy gap between models could simply reflect sampling noise specific to this fixed test set, rather than a genuine, generalizable architectural advantage. To separate real differences from noise, a paired bootstrap resampling procedure was applied: in each of 1,000 iterations, the 850 test images were resampled with replacement, and the same resample was applied to every model in that iteration so that pairing was preserved across models. mAP@50:95 was recomputed for each model on each resample, reusing the already-collected low-confidence-threshold predictions without any re-inference. This produced an empirical distribution of 1,000 mAP@50:95 values per model, from which 95% confidence intervals were derived using the percentile method, as summarized in Table 6. Although RF-DETR's confidence interval overlaps slightly with YOLOv12m's upper bound in Table 6, the paired bootstrap comparisons in Table 7 confirm that RF-DETR significantly outperforms every other model ($p<0.001$), indicating its lead is unlikely to be due to chance alone. Figure 5 illustrates the same result visually, showing the full bootstrap distribution of mAP@50:95 for each model rather than just the interval endpoints, which makes the separation between RF-DETR and the rest of the field, as well as the tight clustering of RT-DETR-L, YOLO11m, and YOLO26m, immediately apparent. Figure 5 complements Table 6 by showing the full shape of each model's bootstrap distribution rather than just its interval endpoints. For each model, the distribution is drawn as a box spanning the interquartile range, with an orange

line marking the median and a green triangle marking the mean of the 1,000 resampled mAP@50:95 values.

Table 6. Point-estimate mAP@50:95 and 95% bootstrap confidence interval per model (1,000 resamples, seed=0).

Model	mAP@50:95	95% CI low	95% CI high
RF-DETR (Medium)	0.682	0.668	0.699
YOLOv12m	0.663	0.647	0.680
YOLO11m	0.654	0.638	0.674
RT-DETR-L *	0.654	0.637	0.672
YOLO26m	0.653	0.637	0.673

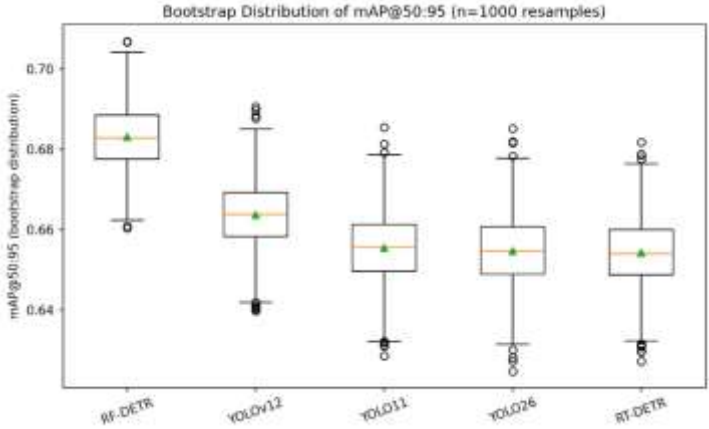


Figure 5: Bootstrap distribution of mAP@50:95 per model (1,000 resamples)

The pattern in Table 7 confirms what Table 6's point estimates suggested: RF-DETR's advantage over every other model is statistically significant at $p<0.001$ in all four comparisons, so its higher point-estimate accuracy is not an artifact of this particular test-set sample. Among the remaining four models, YOLOv12m is statistically significantly ahead of RT-DETR-L, YOLO11m, and YOLO26m ($p = 0.012–0.014$), placing it in its own middle tier. By contrast, the three pairwise comparisons among RT-DETR-L, YOLO11m, and YOLO26m all return large p-values (0.696–0.938), meaning their small point-estimate differences (0.6526–0.6544 mAP@50:95) are statistically indistinguishable from noise. These three models should therefore

be treated as a single, tied bottom tier, and their raw ranking in Table 2 should not be over-interpreted as evidence of architectural superiority. It is worth noting that RF-DETR's confidence interval in Table 6 does overlap slightly with YOLOv12m's, yet the paired test in Table 7 still resolves the difference as significant. This is expected: comparing separate intervals discards the pairing information, whereas the paired bootstrap evaluates both models on the identical resample at every iteration and is therefore the more sensitive test.

Table 7. Pairwise statistical comparisons (paired bootstrap, two-sided p-value from win rate).

Comparison	Win rate	p-value	Significant (p<0.05)	Mean diff.
RF-DETR vs. RT-DETR-L	1.000	<0.001	Yes	+0.029
RF-DETR vs. YOLO11m	1.000	<0.001	Yes	+0.028
RF-DETR vs. YOLOv12m	1.000	<0.001	Yes	+0.019
RF-DETR vs. YOLO26m	1.000	<0.001	Yes	+0.028
RT-DETR-L vs. YOLO11m	0.348	0.696	No	−0.001
RT-DETR-L vs. YOLOv12m	0.007	0.014	Yes	−0.010
RT-DETR-L vs. YOLO26m	0.469	0.938	No	−0.001
YOLO11m vs. YOLOv12m	0.006	0.012	Yes	−0.008
YOLO11m vs. YOLO26m	0.595	0.810	No	+0.001
YOLOv12m vs. YOLO26m	0.994	0.012	Yes	+0.009

To complement the aggregate metrics, the detections produced by each model were inspected visually on five representative test images, at the same confidence = 0.25 operating point used for the precision/recall/F1 results in Table 3. In each panel, the ground-truth plate is drawn as a dashed green box and each model's predicted box, where one is produced, as a solid red box annotated with its confidence score. The five cases were selected to span the range of conditions present in the test set: a typical medium-sized plate (Figure 6), the single smallest plate in the split (Figure 7), a small-object case where two of the five models fail (Figure 8), an unusually large close-up where three of the five fail (Figure 9), and an image containing more than one plate (Figure 10).

These five qualitative examples reinforce, at the level of individual images, the same picture that the aggregate metrics establish numerically. On typical, medium-sized, single-plate images (Figure 6), all five models detect the plate with high confidence and high IoU, visually confirming why the mAP@50 scores in Table 2 are so uniformly high (all above 0.96) — at this common operating condition, architecture simply does not matter much.

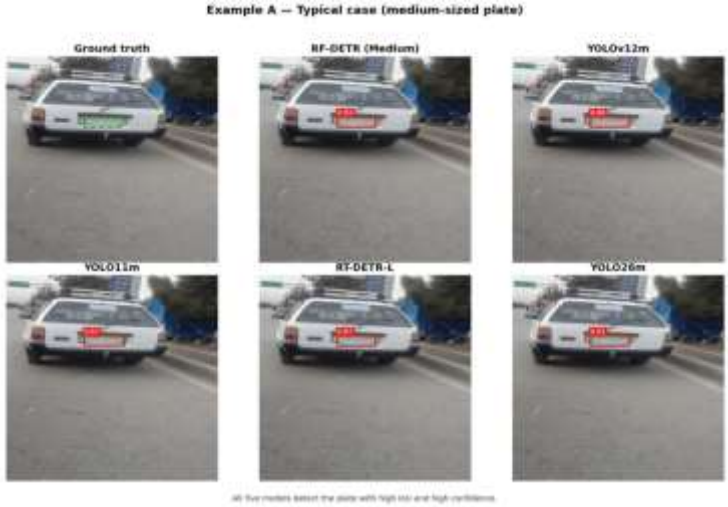


Figure 6: Example A — typical case (medium-sized plate).

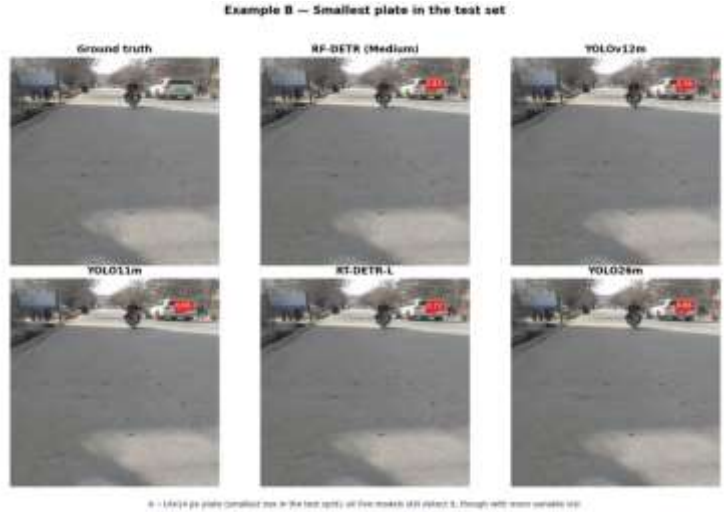


Figure 7: Example B — the smallest plate in the test set.

Figure 7 shows the smallest ground-truth plate in the entire test split, roughly 14 x 14 pixels. All five models detect it, although with visibly more variable confidence and localisation than on medium-sized plates. This is an

important control: it demonstrates that the failures examined below are not simply a function of absolute object size, since every model succeeds here at the smallest scale the dataset contains. The failures that follow are therefore better understood as arising from context, contrast and framing rather than from pixel count alone.

The picture changes at the extremes of object scale, and this is precisely where the aggregate numbers start to diverge. Figure 8 shows a small, distant plate that RF-DETR, YOLO11m, and YOLOv12m detect at moderate confidence (0.48–0.62), while RT-DETR-L and YOLO26m miss it entirely at the confidence=0.25 operating threshold. This single failure is a concrete instance of the pattern already visible in Table 2's AP_small column, where RF-DETR (0.382) and YOLOv12m (0.384) lead and RT-DETR-L (0.352) trails — the qualitative example puts a face on what would otherwise be an abstract half-point difference in AP.

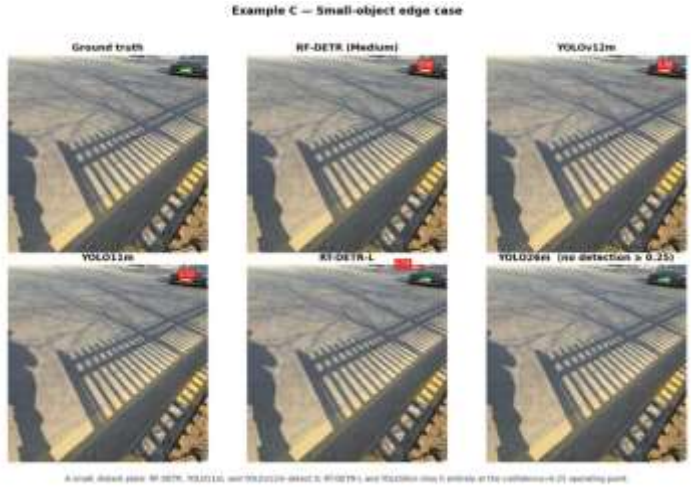


Figure 8: Example C — a small-object edge case.

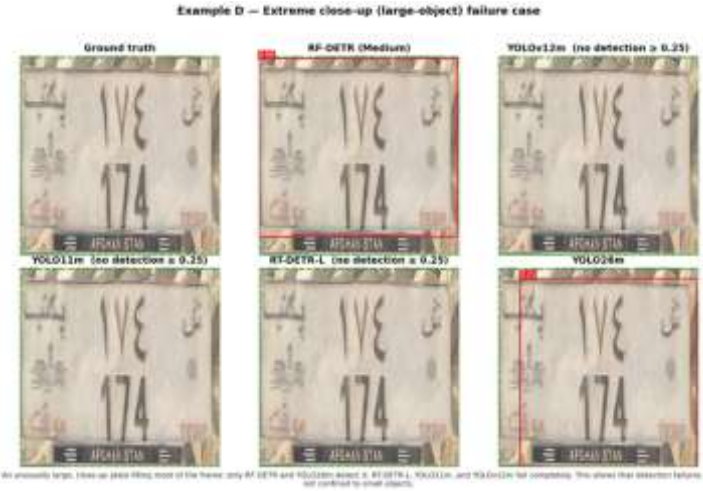


Figure 9: Example D — an extreme close-up (large-object) failure case.

More strikingly, Figure 9 shows the opposite failure mode: an unusually large, close-up plate that fills most of the frame. Here it is RT-DETR-L, YOLO11m, and YOLOv12m that fail completely, while only RF-DETR (0.96 confidence) and YOLO26m (0.25, barely clearing the operating threshold) succeed. This example is important precisely because it is not explained by AP_small at all it is a large-object case which means detection failures on this dataset are not confined to small objects, as the small/medium/large AP breakdown in Table 2 might suggest in isolation. Instead, Figure 9 reveals an out-of-distribution generalization gap: RF-DETR and YOLO26m evidently generalize better to extreme close-up views that fall outside the typical range of plate sizes seen during training, while the other three models do not. This is a failure mode invisible in the summary tables alone, since Table 2 only reports AP at three fixed size bins and cannot capture generalization to atypical scales within those bins yet it is directly relevant to real-world deployment, where camera distance and zoom are rarely fixed or fully controlled.

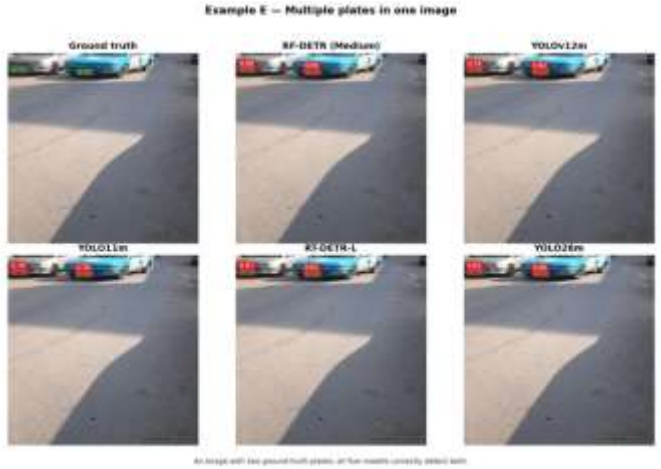


Figure 10: Example E — multiple plates in one image.

Figure 10 shows an image containing two ground-truth plates, both of which are correctly detected by all five models. Beyond confirming that none of the architectures is limited to single-object scenes, this case illustrates the AR@10 and AR@100 equivalence noted in Table 5: with at most a small number of plates per image, the detection cap imposed by AR@10 is never reached, and the two columns therefore measure the same quantity on this dataset.

CONCLUSION

All five models reach very high mAP@50 scores (above 0.96), indicating that single-class license plate detection is largely "solved" by current-generation architectures, with the real differentiation emerging on the stricter mAP@50:95 threshold and small-object performance, where RF-DETR (Medium) leads with a statistically significant margin (mAP@50:95=0.682) at the cost of the longest training time (~5.9 hours) and one of the lowest inference speeds (29.0 FPS), reflecting the typical accuracy-for-compute trade-off of transformer-based DETR architectures; YOLOv12m follows as a statistically validated second tier (mAP@50:95=0.663) with the highest VRAM footprint among the YOLO models (20.1 GB), while RT-DETR-L, YOLO11m, and YOLO26m form a third, statistically indistinguishable tier (mAP@50:95 between 0.653–0.654) that nonetheless differs on secondary axes — YOLO26m and YOLO11m lead on speed and low memory use (up to 46.8 FPS, as low as 14.4 GB VRAM),

while RT-DETR-L uniquely leads all models on small-object recall (AR_small=0.655) despite needing AMP disabled during training and the largest VRAM footprint (34.5 GB), a training exception noted here for transparency.

Practically, YOLO26m or YOLO11m are preferable for near-real-time, memory-constrained deployment, RF-DETR (Medium) for accuracy-prioritized scenarios with a larger compute budget, and YOLOv12m as a validated middle ground.

These conclusions should, however, be read with several limitations in mind the benchmark rests on a single dataset and single object class, inference speed is specific to the RTX 6000 Pro Blackwell hardware and batch=1 setting, RT-DETR-L's AMP-disabled training may bias its result conservatively, the small/medium/large size thresholds follow COCO's generic definitions rather than plate-specific optimization, and the reported confidence intervals capture only test-set sampling uncertainty, not training-run variance, since each model was trained with a single seed.

REFERENCES

Ahmadi, M., Fazel, T., & Algaradi, K. (2026). End-to-end Afghan license plate detection and recognition using deep learning models (Version 1) [Data set]. Mendeley Data, https://doi.org/10.17632/7vzmbpg296.1

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. (2020). End-to-end object detection with transformers. Proceedings of the European Conference on Computer Vision (ECCV), 213–229.

Ismail, A., Mehri, M., Sahbani, A., & Essoukri Ben Amara, N. (2025). Automatic license plate recognition in in-the-wild scenarios: A comprehensive review, open issues, and future directions. IEEE Access, 13, 145387–145415.

Jocher, G., Qiu, J., & Chaurasia, A. (2023). Ultralytics YOLO (Version 8.0.0) [Computer software]. https://github.com/ultralytics/ultralytics

Laroca, R., Estevam, V., Moreira, G. J. P., Minetto, R., & Menotti, D. (2025). Advancing multinational license plate recognition through synthetic and real data fusion: A comprehensive evaluation. IET Intelligent Transport Systems, 19(1), e70086.

Zhao, Y., Lv, W., Xu, S., Wei, J., Wang, G., Dang, Q., ... & Chen, J. (2024, June). Detrs beat yolos on real-time object detection. In 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 16965-16974). IEEE.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. Proceedings of the European Conference on Computer Vision (ECCV), 740–755.

Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Assran, M., Ballas, N., Galuba, W., Howes, R., Huang, P.-Y., Li, S.-W., Misra, I., Rabbat, M., Sharma, V., … Bojanowski, P. (2023). DINOv2: Learning robust visual features without supervision. arXiv preprint, arXiv:2304.07193.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 779–788.

Robinson, I., Robicheaux, P., Popov, M., Ramanan, D., & Peri, N. (2025). RF-DETR: Neural architecture search for real-time detection transformers. arXiv preprint, arXiv:2511.09554.

Tian, Y., Ye, Q., & Doermann, D. (2025). YOLOv12: Attention-centric real-time object detectors. Advances in Neural Information Processing Systems (NeurIPS), 38, 78433–78457.

Xu, L., Shang, Z., Chen, X., Zeng, T., Dai, W., & Zhao, J. (2026). Deep learning algorithms for license plate recognition: A review. Neural Networks, 108842.

Zhao, Z.-Q., Zheng, P., Xu, S.-T., & Wu, X. (2019). Object detection with deep learning: A review. IEEE Transactions on Neural Networks and Learning Systems, 30(11), 3212–3232.

Modern Image Processing and Computer Vision: From Classical Techniques to Intelligent Visual Analysis

Muhammed Mücahit ARVAS

Yazılım Mühendisliği Bölümü, Iğdır Üniversitesi, Iğdır, Türkiye E-mail: mmucahit.arvas@igdir.edu.tr

1. INTRODUCTİON

Digital images have become one of the most important forms of information in contemporary computing systems. The widespread availability of smartphones, surveillance cameras, medical imaging devices, unmanned aerial vehicles, satellites, industrial inspection systems, and autonomous platforms has resulted in the continuous generation of large volumes of visual data. However, the acquisition of an image alone does not necessarily provide meaningful information. In most practical applications, images must be processed, enhanced, analyzed, and interpreted before they can support reliable decisions. Image processing therefore represents a fundamental stage between raw visual data acquisition and higher-level visual understanding.

In its classical definition, digital image processing refers to computational operations applied to digital images in order to improve their visual quality, extract meaningful characteristics, suppress undesirable components, or transform the image into a representation suitable for subsequent analysis. A digital image can mathematically be represented as a two-dimensional discrete function $I(x, y)$, where $x$and $y$denote spatial coordinates and the corresponding function value represents the intensity or color information associated with a particular pixel (Szeliski, 2022). Depending on the acquisition system, an image may consist of a single grayscale channel, multiple color channels, spectral bands, depth information, or other modality-specific measurements.

The development of image-processing methodologies has historically progressed from relatively simple pixel-level transformations toward increasingly sophisticated computational representations. Traditional techniques such as intensity transformation, histogram modification, spatial filtering, thresholding, edge detection, mathematical morphology, and geometric transformation remain widely used because of their computational efficiency, interpretability, and limited data requirements. These methods are particularly valuable when the characteristics of the target objects and image-degradation mechanisms are sufficiently understood in advance (Münke et al., 2024).

Nevertheless, modern imaging environments frequently involve complex conditions such as non-uniform illumination, sensor noise, motion blur, occlusion, background variability, scale differences, and highly heterogeneous object appearances. Such challenges have encouraged the

development of learning-based approaches capable of automatically extracting informative visual representations from data. Recent studies have demonstrated that deep convolutional neural networks (CNNs) have substantially advanced image classification, object detection, semantic segmentation, super-resolution, and related computer-vision tasks by learning hierarchical representations directly from image data (Zhao et al., 2024). At the same time, conventional image-processing methods remain highly relevant, particularly in applications characterized by limited training data, constrained computational resources, or strong prior knowledge regarding image structure (Münke et al., 2024).

Consequently, contemporary image analysis should not be considered as a strict competition between traditional processing and artificial intelligence. Instead, modern computer-vision pipelines frequently integrate both approaches. Classical operations may be used to remove noise, correct illumination, normalize intensity distributions, enhance local contrast, or isolate regions of interest before information is provided to a machine-learning or deep-learning model. Such hybrid pipelines can improve data quality while reducing unnecessary computational complexity (Münke et al., 2024; Zhao et al., 2024).

Recent developments have further expanded this paradigm. Transformer-based architectures have introduced self-attention mechanisms capable of modelling long-range relationships between image regions and have become increasingly influential in visual recognition and interpretation. Vision Transformer variants are now used in classification, detection, segmentation, generation, and other visual-analysis tasks, while considerable research focuses on reducing their computational requirements for practical deployment (Papa et al., 2024). These developments illustrate the transition from manually designed visual descriptors toward increasingly general-purpose representation-learning systems.

Image enhancement has experienced a similar evolution. Classical techniques including contrast stretching, histogram equalization, spatial filtering, and Retinex-based processing continue to provide efficient solutions for many applications, whereas recent learning-based approaches increasingly address complex degradations that cannot easily be modelled using fixed mathematical operations. Current research therefore treats image enhancement as an important bridge between low-level visual processing and higher-level computer-vision performance (Lepcha et al., 2023).

Image processing and computer vision now support a broad range of scientific and industrial applications. In healthcare, visual-analysis algorithms assist with the interpretation of radiological and microscopic images. In agriculture, images acquired from field cameras, unmanned aerial vehicles, and satellites can be analyzed for crop monitoring and disease detection. Industrial systems employ machine vision for surface-defect inspection, quality control, dimensional measurement, and automated manufacturing. Similarly, intelligent transportation systems use image analysis for vehicle detection, traffic monitoring, lane analysis, and autonomous navigation. Remote sensing, robotics, security, cultural-heritage documentation, environmental monitoring, and infrastructure inspection constitute additional important application domains.

As illustrated in Figure 1, a contemporary visual-analysis system can be considered as a layered workflow beginning with image acquisition and digital representation. Low-level processing operations subsequently improve image quality and suppress undesirable components. Intermediate processing stages extract structures such as edges, regions, textures, and candidate objects. Finally, high-level computer-vision models transform these representations into semantic outputs such as object identities, spatial locations, segmentation masks, or automated decisions. Importantly, the boundaries between these stages are increasingly flexible because modern learning-based architectures may jointly perform several operations within a single end-to-end framework.

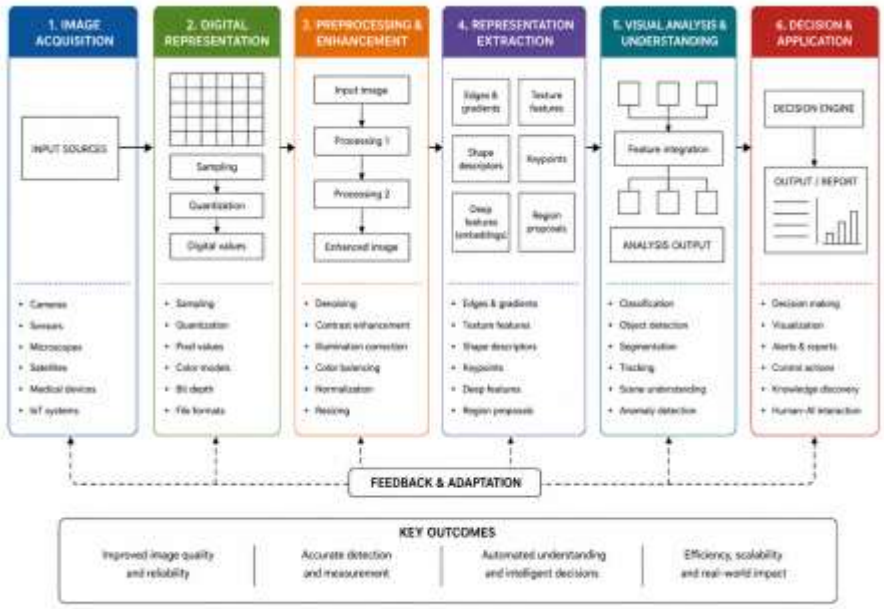


Figure 1. General workflow of a modern image-processing and computer-vision system, progressing from image acquisition and preprocessing to representation extraction, visual analysis, and decision-making.

The proposed representation in Figure 1 emphasizes that image preprocessing should not be regarded merely as an isolated manipulation of pixel values. Instead, preprocessing constitutes an integral component of the overall visual-analysis pipeline. The quality of the acquired image, the selected representation, and the preprocessing strategy may directly influence the reliability of subsequent segmentation, feature extraction, classification, detection, and interpretation stages. This relationship becomes particularly important when visual data are collected under uncontrolled environmental conditions.

Another important development is the increasing interaction between local and global visual representations. Classical filters and convolutional neural networks generally emphasize spatially local relationships, whereas attention-based approaches can explicitly model interactions between distant image regions. This difference has motivated extensive investigation of CNNs, Vision Transformers, and hybrid architectures. Current evidence indicates that the optimal solution depends strongly on data availability, application

requirements, computational constraints, and the structural characteristics of the visual information being analyzed (Zhao et al., 2024; Papa et al., 2024).

Accordingly, understanding contemporary image processing requires knowledge of both fundamental mathematical operations and modern learning-based visual-analysis strategies. The objective of this chapter is to provide a structured overview of this continuum. First, the fundamental principles of digital image representation, pixels, intensity values, and color spaces are introduced. Subsequently, major preprocessing and enhancement techniques including intensity transformations, histogram-based methods, noise filtering, edge-preserving operations, and image resizing are discussed. Thresholding, edge analysis, segmentation, and mathematical morphology are then examined as important mechanisms for extracting meaningful structures from images. Finally, the chapter discusses the transition from conventional image-processing pipelines to modern deep-learning-based computer vision, together with representative application areas, current limitations, and emerging research directions.

Rather than treating traditional and modern methods as independent concepts, this chapter highlights their complementary roles within contemporary visual-analysis systems. Such an integrated perspective provides a more realistic representation of current practice, where carefully designed preprocessing operations and learned visual representations are frequently combined to improve robustness, efficiency, and interpretability.

2. FUNDAMENTALS OF DIGITAL IMAGE REPRESENTATION

Digital image processing begins with the conversion of visual information into a numerical form that can be interpreted and processed by a computer. Images obtained from cameras, medical imaging devices, satellites, industrial inspection systems, microscopes, or other sensors are represented as discrete data structures composed of pixels. The way in which these pixels are organized, sampled, quantized, and encoded directly affects subsequent processing stages such as enhancement, segmentation, feature extraction, classification, and object detection (Szeliski, 2022).

A digital image can be defined as a two-dimensional numerical representation in which each spatial location corresponds to a pixel value. For a grayscale image with M rows and N columns, a compact representation can be written as:

$$I = [I(x,y)]_{M \times N}$$

where I denotes the digital image, x and y represent spatial coordinates, and I(x,y) indicates the intensity value of the pixel at a specific location.

In color images, each spatial position is represented by more than one numerical value. For example, an RGB image includes three separate channels corresponding to red, green, and blue information. These representations form the basis of nearly all conventional and modern computer-vision applications.

2.1. Pixels, Sampling, and Quantization

The pixel is the smallest addressable element of a digital image. Each pixel contains numerical information associated with a particular spatial position. The total number of pixels determines the spatial dimensions of the image and influences the amount of visual detail that can be represented.

The transformation of a continuous visual scene into a digital image fundamentally involves two operations: sampling and quantization. Sampling determines how the continuous spatial domain is divided into discrete positions, while quantization assigns numerical intensity values to those sampled positions.

Sampling is closely related to spatial resolution. A higher sampling density generally allows smaller structures and finer details to be represented more accurately. However, increasing the number of pixels also increases storage requirements and computational cost. Therefore, the required image resolution should be selected according to the characteristics of the target application.

Quantization determines the number of intensity levels available for representing each pixel. If a pixel is represented using b bits, the total number of possible intensity levels is:

$$L = 2^b$$

where L represents the number of intensity levels and b denotes bit depth.

For example, an 8-bit grayscale image contains:

$$L = 2^8 = 256$$

different intensity levels.

These values generally range from 0 to 255, where lower values correspond to darker regions and higher values represent brighter regions. Increasing bit depth allows smaller intensity variations to be represented, which can be important in medical imaging, remote sensing, scientific imaging, and industrial inspection. However, higher bit depth also increases storage and computational requirements.

The overall relationship among scene acquisition, sampling, quantization, and digital image formation is illustrated in Figure 2.

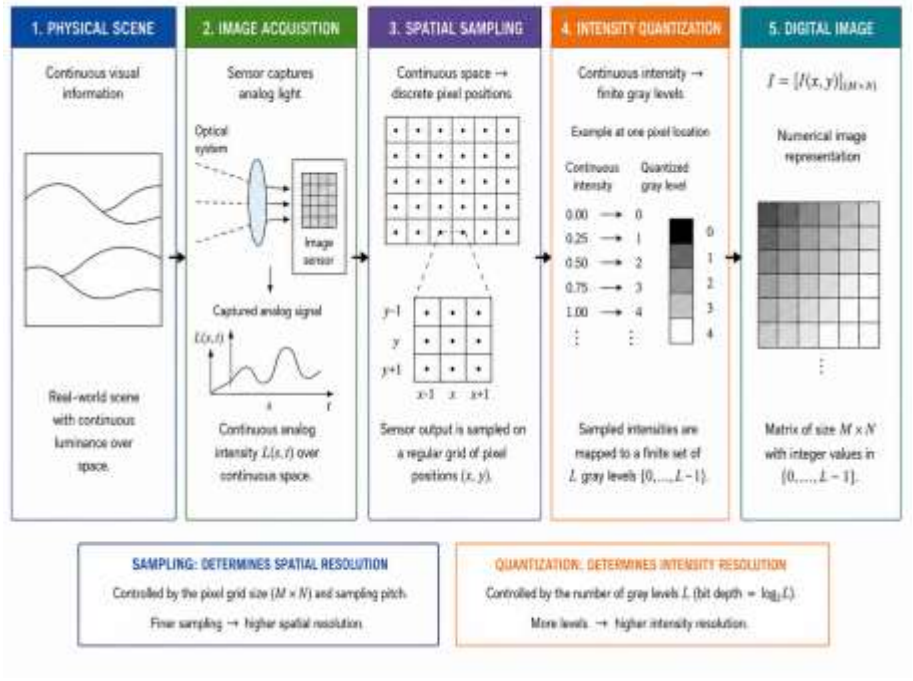


Figure 2. Conceptual process of digital image formation through image acquisition, spatial sampling, and intensity quantization.

As shown in Figure 2, visual information is first captured from a physical scene, then spatially sampled and numerically quantized before being converted into a digital image. These operations establish the basic numerical structure required for subsequent image-processing applications.

2.2. Grayscale Image Representation

A grayscale image represents visual information using a single intensity channel. Each pixel contains one numerical value that indicates its brightness level.

In a typical 8-bit grayscale image, pixel values vary between 0 and 255. A value of 0 corresponds to black, while 255 corresponds to white. Intermediate values represent different shades of gray.

Grayscale representation is commonly used because many image-processing operations do not require complete color information. Thresholding, edge detection, texture analysis, contour extraction, and several filtering methods can be performed effectively using grayscale images.

Color images can be converted into grayscale by combining the red, green, and blue components. A commonly used weighted transformation is:

$$\mathbf{Y = 0.299R + 0.587G + 0.114B}$$

where R, G, and B denote the red, green, and blue channel values, while Y represents the resulting grayscale intensity.

The coefficients are weighted differently because human visual perception is more sensitive to some wavelengths than others. Therefore, weighted grayscale conversion generally provides a more realistic luminance representation than a simple arithmetic average (Szeliski, 2022).

2.3. Binary Image Representation

A binary image contains only two possible pixel states. These values are commonly represented as 0 and 1 or, in 8-bit image-processing environments, as 0 and 255.

Binary images are frequently used to separate foreground objects from the background. They are especially important in segmentation, contour detection, connected-component analysis, character recognition, shape analysis, and mathematical morphology.

A simple binary transformation can be represented as:

$$\mathbf{B(x,y) = 1, \text{ if } I(x,y) \geq T}$$

B(x,y) = 0, if I(x,y) < T

where T denotes the selected threshold value.

Pixels satisfying the threshold condition are assigned to the foreground class, while the remaining pixels are assigned to the background class. The quality of binary separation therefore depends strongly on the selected threshold value.

A fixed global threshold may be sufficient when object and background intensities are clearly separated. In more complex images, adaptive or automatically determined thresholds are often more appropriate. These approaches will be discussed in the following sections.

2.4. RGB Color Representation

The RGB color model is one of the most widely used representations for digital color images. In this model, each pixel is represented by three components corresponding to red, green, and blue.

An RGB pixel can be expressed as:

I(x,y) = [R(x,y), G(x,y), B(x,y)]

where R(x,y), G(x,y), and B(x,y) denote the values of the three color channels at the same spatial position.

In a standard 8-bit-per-channel image, each channel can contain values between 0 and 255. Therefore, a 24-bit RGB image can theoretically represent:

256 × 256 × 256 = 16,777,216 colors

RGB representation is widely used in digital cameras, smartphones, display devices, and deep-learning systems. Nevertheless, RGB may not always provide the most suitable representation for a particular image-analysis problem because illumination variations can simultaneously affect all three color channels.

For this reason, alternative color spaces are commonly used to reorganize color information according to the requirements of the analysis task.

2.5. Alternative Color Spaces

Different color spaces describe visual information using different component organizations. The most commonly used alternatives to RGB include HSV and CIE Lab.

The HSV color space consists of:

Hue (H): dominant color information
Saturation (S): color purity
Value (V): brightness information

The main advantage of HSV is that color characteristics can be partially separated from brightness. This makes HSV useful in color-based segmentation, object tracking, and applications affected by moderate illumination changes.

The CIE Lab color space includes:

L* = lightness component
a* = green-to-red chromatic component
b* = blue-to-yellow chromatic component

Lab representation is especially useful when perceptual color differences are important. It is therefore commonly used in color comparison, industrial inspection, image enhancement, clustering, and segmentation.

Modern image-processing libraries provide direct conversion among these image representations. OpenCV supports conversion between grayscale, RGB/BGR, HSV, Lab, XYZ, and several other color spaces.

The main characteristics of commonly used image representations are summarized in Table 1.

Table 1. Comparison of commonly used digital image representations in image processing and computer vision.

Representation	Components	Main Characteristic	Typical Applications
Binary	Two pixel states	Foreground-background separation	Morphology, masks, connected components
Grayscale	Single intensity channel	Low computational complexity	Filtering, thresholding, edge detection
RGB	Red, Green, Blue	Direct color representation	Acquisition, visualization, deep learning
HSV	Hue, Saturation, Value	Partial separation of color and brightness	Color segmentation, object tracking
CIE Lab	L*, a*, b*	Perceptually oriented color representation	Color analysis, enhancement, clustering

As presented in Table 1, each representation has different advantages. Grayscale and binary images reduce computational complexity, RGB preserves complete color information, and HSV or Lab may be preferred when color characteristics need to be analyzed more independently from illumination (Zangana et al., 2024).

2.6. Spatial Resolution, Intensity Resolution, and Dynamic Range

Image quality is not determined only by the number of pixels. Spatial resolution, intensity resolution, dynamic range, noise level, illumination quality, optical focus, and compression may all influence the amount of useful information available in an image.

Spatial resolution indicates the level of spatial detail that can be represented. Higher spatial resolution may improve the visibility of small structures but also increases memory use and computational cost.

Intensity resolution is related to the number of available brightness levels. Greater intensity resolution allows smaller variations in signal magnitude to be represented.

Dynamic range describes the difference between the lowest and highest measurable signal levels. Limited dynamic range may result in loss of information in very dark or very bright image regions.

These properties are particularly important in modern computer-vision systems. For example, excessive image downsampling may remove small but meaningful structures, while unnecessarily large image dimensions may increase inference time without providing a significant accuracy improvement.

Therefore, image representation should always be selected according to both the visual characteristics of the data and the computational requirements of the target application.

2.7. Role of Image Representation in Modern Computer Vision

Traditional image-processing algorithms generally operate directly on pixel intensities or on manually defined image characteristics such as edges, textures, corners, and contours. Modern deep-learning methods extend this approach by automatically learning increasingly complex feature representations from image data.

Although deep-learning systems can learn high-level features automatically, the quality of the input image remains important. Incorrect channel ordering, inappropriate normalization, excessive resizing, poor contrast, or unsuitable color conversion may negatively influence the performance of the entire model.

For this reason, fundamental image-representation concepts remain relevant even in advanced artificial-intelligence applications. The principles of sampling, quantization, bit depth, spatial resolution, grayscale conversion, and color-space selection continue to form the basis of reliable visual-analysis systems.

Accordingly, the choice of image representation should be guided by the information required by the downstream task. Grayscale representations may simplify structural analysis, HSV or Lab can facilitate color-based processing, and binary images provide an efficient format for segmentation and morphological operations. These principles establish the foundation for the image-enhancement and preprocessing techniques discussed in the following section.

3. IMAGE ENHANCEMENT AND PRE-PROCESSING TECHNIQUES

Image preprocessing represents an essential stage in many image-processing and computer-vision pipelines. Images acquired under real-world conditions may contain noise, low contrast, non-uniform illumination, blur, color distortion, compression artifacts, or unwanted background variations. These degradations can reduce visual quality and may also negatively affect subsequent operations such as segmentation, feature extraction, classification, and object detection. Therefore, preprocessing techniques aim to improve the quality and consistency of image data before higher-level analysis is performed.

Image enhancement does not necessarily attempt to reconstruct an ideal version of the original scene. Instead, its primary objective is to emphasize useful visual information while suppressing irrelevant or undesirable components. Depending on the application, this may involve contrast adjustment, intensity normalization, histogram modification, noise reduction, smoothing, sharpening, or resolution transformation. Recent research increasingly combines conventional enhancement methods with learning-based techniques, particularly when image degradation is complex or difficult to describe analytically (Lepcha et al., 2023).

3.1. Intensity Transformation and Normalization

Intensity transformation modifies pixel values in order to improve visibility or standardize image characteristics. One of the simplest approaches is linear contrast stretching, in which the original intensity interval is mapped to a wider output range.

A normalized pixel value can be expressed as:

$$I_n = (I - Imin) / (Imax - Imin)$$

where:

I = original pixel intensity
Imin = minimum intensity in the image
Imax = maximum intensity in the image
$\mathbf{I_n}$ = normalized pixel intensity

This transformation maps pixel values into a standardized interval, typically between 0 and 1. Normalization is widely used before machine-learning and deep-learning operations because it reduces differences caused by numerical scale and provides more consistent model inputs.

Another commonly used transformation is gamma correction. Gamma transformation modifies image brightness according to:

$$\mathbf{Iout = c \times Iin^{\gamma}}$$

where Iin represents the original normalized intensity, Iout represents the transformed intensity, c is a scaling coefficient, and γ is the gamma parameter.

When $\gamma < 1$, darker regions are generally enhanced, while $\gamma > 1$ tends to suppress low-intensity values and produce a darker output. Gamma correction can therefore be useful when images suffer from inappropriate exposure or when particular intensity ranges must be emphasized.

The selection of enhancement parameters should depend on the characteristics of the image. Excessive intensity modification can amplify noise, eliminate subtle structures, or generate unrealistic visual appearances. Enhancement should therefore preserve relevant image information rather than merely producing visually attractive results.

3.2. Histogram Analysis and Contrast Enhancement

An image histogram describes the distribution of pixel intensities. For a grayscale image, the horizontal axis represents intensity levels, while the vertical axis represents the number or frequency of pixels associated with each level.

Histogram analysis provides useful information about overall brightness, contrast, and intensity distribution. A histogram concentrated within a narrow intensity interval generally indicates limited contrast, whereas a broader distribution usually represents greater separation among intensity levels.

Histogram equalization is a classical enhancement technique designed to redistribute image intensities over a broader range. The transformation is derived from the cumulative distribution of the original histogram and can increase the visibility of structures in low-contrast images.

Global histogram equalization applies a single transformation to the entire image. Although computationally efficient, this approach may be unsuitable when different regions of an image are affected by different illumination conditions. Local enhancement strategies can provide better results in such situations.

One of the most widely used local methods is Contrast Limited Adaptive Histogram Equalization (CLAHE). Instead of processing the entire image as a single region, CLAHE divides the image into smaller local areas and performs contrast enhancement within each area while limiting excessive amplification. This restriction is particularly important because ordinary adaptive histogram equalization may strongly amplify local noise.

Recent image-enhancement research continues to treat histogram-based techniques as useful conventional methods because of their simplicity, interpretability, and low computational requirements. At the same time, learning-based enhancement approaches are increasingly used when low illumination, color distortion, and noise occur simultaneously (Lepcha et al., 2023).

The main contrast-enhancement strategies discussed above are summarized conceptually in Figure 3.

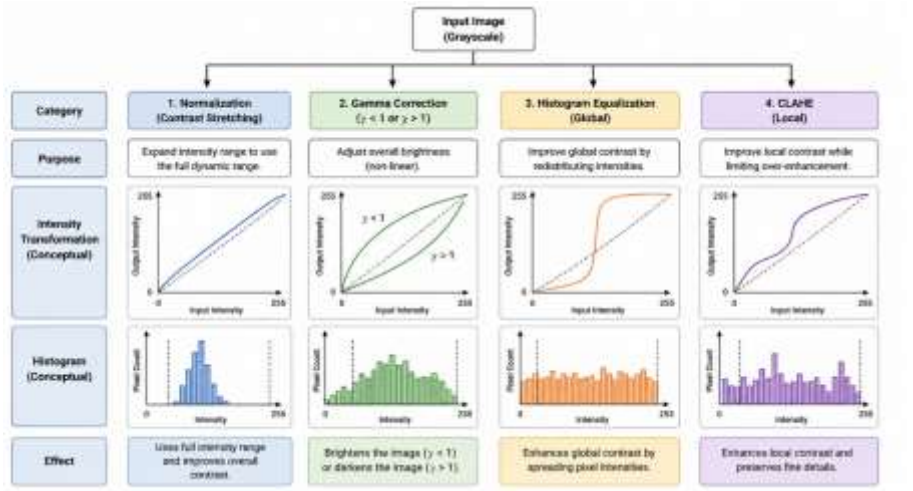


Figure 3. General comparison of intensity and contrast-enhancement strategies, including normalization, gamma correction, global histogram equalization, and locally adaptive histogram enhancement.

As illustrated in Figure 3, different enhancement techniques modify intensity information according to different principles. Normalization adjusts the numerical range, gamma correction changes the nonlinear relationship among intensity levels, histogram equalization redistributes the global intensity distribution, and CLAHE performs controlled enhancement at a local level.

3.3. Noise and Image Degradation

Noise represents unwanted variations in pixel values that are unrelated to the actual scene structure. It may originate from electronic sensors, illumination conditions, transmission errors, environmental interference, compression, or image reconstruction procedures.

Different noise models require different filtering approaches. Gaussian noise is generally characterized by continuous random intensity fluctuations, whereas salt-and-pepper noise produces isolated extreme pixel values. Other image-degradation mechanisms may include speckle noise, motion blur, defocus blur, haze, or low-light artifacts.

Noise removal requires a balance between suppression and structural preservation. Aggressive smoothing may reduce noise but can also remove edges, textures, and small objects. This trade-off is particularly important in

scientific and computer-vision applications because small structures may carry essential information.

A recent comprehensive review of digital image denoising emphasizes that conventional algorithms remain effective for relatively simple noise patterns because of their low computational cost and straightforward implementation. However, deep neural networks have become increasingly important for complex and high-level noise because they can learn relationships between image structures and degradation patterns directly from data (Mao et al., 2025).

3.4. Mean, Gaussian, and Median Filtering

Spatial-domain filtering modifies a pixel according to information obtained from a local neighborhood surrounding that pixel. A small matrix, commonly referred to as a kernel or filter window, is moved across the image and used to generate the output.

The mean filter replaces a pixel by the average intensity within a local neighborhood. For a 3 × 3 neighborhood, this operation can reduce random variations but generally smooths both noise and structural details. As a result, edges may become blurred.

Gaussian filtering provides weighted smoothing. Pixels closer to the center of the kernel receive larger weights than distant pixels. This approach generally produces smoother and more natural results than uniform averaging and is widely used as a preprocessing stage before edge detection and feature extraction.

Median filtering differs from mean and Gaussian filtering because the output value is obtained by sorting the intensities inside the neighborhood and selecting the median value. It is particularly effective for removing impulse or salt-and-pepper noise while preserving sharp boundaries.

For example, if the neighborhood contains the values:

12, 14, 15, 16, 17, 18, 21, 23, 255

the median value is:

17

Therefore, the extreme value 255 does not dominate the output pixel, as it would in a simple averaging operation.

3.5. Edge-Preserving Filtering

Conventional smoothing filters may remove both noise and meaningful edges. Edge-preserving filters were developed to reduce this problem by considering not only spatial distance but also differences among pixel intensities.

The bilateral filter is a representative example. It combines spatial weighting with intensity similarity so that neighboring pixels with similar intensities contribute more strongly to the output than pixels located across an edge. Consequently, homogeneous regions can be smoothed while major boundaries are preserved.

Such filters are useful when preprocessing must reduce noise without destroying structural details required by segmentation or recognition algorithms. Nevertheless, bilateral filtering generally requires greater computational effort than simple mean or Gaussian filtering.

More advanced denoising approaches, such as non-local filtering and transform-domain methods, exploit repeated image structures or frequency-domain characteristics. Modern deep-learning-based methods extend this idea further by learning degradation patterns directly from image datasets.

3.6. Sharpening and Detail Enhancement

While smoothing reduces high-frequency variations, sharpening is used to emphasize edges and local transitions. Sharpening techniques can improve the apparent clarity of boundaries and fine details but may also amplify noise if applied excessively.

A common sharpening principle is based on combining the original image with an extracted high-frequency component. Conceptually, the process can be represented as:

$$\mathbf{I_{sharp} = I + k \times H}$$

where:

I = original image
H = high-frequency detail component

k = sharpening strength
Isharp = sharpened output image

The value of k determines the degree of enhancement. Large values may increase edge visibility but can also produce ringing, noise amplification, or unnatural boundaries.

Therefore, sharpening should generally be applied after appropriate noise reduction rather than directly to severely degraded images.

3.7. Image Resizing and Interpolation

Resizing is frequently required because image-processing algorithms and deep-learning models often expect specific input dimensions. Image dimensions may be reduced to decrease computational requirements or increased to satisfy visualization and processing requirements.

However, resizing changes the spatial arrangement of image samples. Interpolation is therefore required to estimate pixel values at new positions.

Common interpolation strategies include nearest-neighbor, bilinear, and bicubic interpolation. Nearest-neighbor interpolation is computationally simple but may produce block-like artifacts. Bilinear interpolation estimates new values using nearby pixels and usually generates smoother outputs. Bicubic interpolation considers a larger neighborhood and generally provides better visual quality at the expense of additional computational cost.

Downsampling requires particular attention because fine structures may disappear when image resolution is reduced. This can be problematic in applications involving small objects, thin boundaries, microscopic structures, or subtle abnormalities.

Super-resolution techniques attempt to reconstruct higher-resolution imagery from lower-resolution inputs. Traditional interpolation methods provide computationally efficient solutions, while contemporary deep-learning-based super-resolution models can learn complex mappings between low- and high-resolution data. Recent reviews highlight the increasing role of such approaches in medical imaging, remote sensing, surveillance, and other computer-vision applications (Lepcha et al., 2023).

3.8. Selection of Pre-processing Techniques

No single preprocessing method is optimal for every image or application. The appropriate operation depends on the degradation characteristics of the data and the requirements of subsequent analysis.

Table 2 summarizes the primary characteristics of commonly used preprocessing techniques.

Table 2. Comparison of commonly used image preprocessing and enhancement techniques.

Method	Main Purpose	Main Advantage	Main Limitation
Normalization	Standardize intensity range	Simple and computationally efficient	Does not directly remove noise
Gamma correction	Adjust brightness distribution	Effective for exposure correction	Requires appropriate gamma selection
Histogram equalization	Improve global contrast	Simple and automatic	May over-enhance noise
CLAHE	Improve local contrast	Better for non-uniform illumination	Parameter dependent
Mean filter	Reduce random variations	Very simple	Blurs edges
Gaussian filter	Smooth image noise	Stable and widely applicable	Fine details may be reduced
Median filter	Remove impulse noise	Preserves edges relatively well	Less effective for some continuous noise

Method	Main Purpose	Main Advantage	Main Limitation
Bilateral filter	Denoise while preserving edges	Maintains major boundaries	Higher computational cost
Sharpening	Enhance edges and details	Improves local visibility	Can amplify noise
Resizing	Modify spatial dimensions	Supports standardized model inputs	May remove or distort details

As summarized in Table 2, preprocessing techniques present different trade-offs between enhancement, noise suppression, detail preservation, and computational complexity. Therefore, preprocessing should not be applied as a fixed sequence without considering image characteristics.

In practical computer-vision systems, several operations are often combined. For example, an image may first be resized and normalized, followed by noise reduction and local contrast enhancement. However, unnecessary preprocessing should be avoided because repeated transformations may remove useful information or introduce artifacts.

Recent research has also demonstrated that image degradation under conditions such as low illumination, haze, and complex sensor noise may require more sophisticated solutions than fixed classical filters. Consequently, modern pipelines increasingly combine conventional enhancement methods with learning-based models while retaining classical operations when computational efficiency, interpretability, and limited-data conditions are important (Lepcha et al., 2023; Mao et al., 2025).

Overall, preprocessing should be regarded as a task-dependent stage designed to improve the reliability of downstream visual analysis rather than merely to improve the visual appearance of an image. Appropriate preprocessing can increase contrast, suppress unwanted noise, preserve structural information, and provide more consistent inputs for both conventional algorithms and modern learning-based computer-vision models.

4. SEGMENTATION, EDGE ANALYSIS, AND MATHEMATICAL MORPHOLOGY

Image enhancement improves visual quality, but many computer-vision applications require a further step in which meaningful structures are separated from the surrounding image content. Segmentation, edge analysis, and mathematical morphology provide fundamental tools for this purpose. These techniques transform raw pixel information into regions, boundaries, shapes, or masks that can be analyzed more efficiently by subsequent algorithms.

Image segmentation can be defined as the process of partitioning an image into meaningful and relatively homogeneous regions. The objective is generally to separate objects of interest from the background or to divide the image into regions with similar intensity, color, texture, or semantic characteristics. Segmentation is therefore an important intermediate stage between low-level preprocessing and high-level visual interpretation.

Classical segmentation techniques include thresholding, region-based approaches, edge-based methods, clustering, and watershed-based processing. More recently, convolutional neural networks and transformer-based architectures have significantly expanded the capabilities of image segmentation, particularly for complex scenes in which manually defined intensity or texture rules are insufficient. Recent reviews indicate that contemporary segmentation research increasingly combines traditional image-processing principles with deep-learning-based dense prediction methods (Brar et al., 2025).

4.1. Threshold-Based Segmentation

Thresholding is one of the simplest and most widely used image-segmentation strategies. The basic principle is to assign pixels to different classes according to their intensity values.

For a grayscale image I(x,y), a simple binary threshold operation can be written as:

$$\mathbf{B(x,y) = 1, \text{ if } I(x,y) \geq T}$$

$$\mathbf{B(x,y) = 0, \text{ if } I(x,y) < T}$$

where T represents the selected threshold value.

The effectiveness of global thresholding depends on the intensity separation between the foreground and background. When both regions have clearly different intensity distributions, a single threshold value may provide satisfactory results. However, global thresholding becomes less reliable when illumination changes across the image or when foreground and background intensity distributions overlap.

Adaptive thresholding addresses this limitation by calculating different threshold values for different local regions. Instead of applying one T value to the complete image, the threshold is determined from the statistical characteristics of a neighborhood surrounding each pixel. This approach is especially useful for images captured under non-uniform illumination.

Thresholding remains important because of its computational simplicity and interpretability. However, its performance is strongly dependent on image quality and intensity distribution.

4.2. Otsu Thresholding

Manual threshold selection may introduce subjectivity and may not generalize well across multiple images. Otsu's method provides an automatic threshold-selection strategy based on image statistics (Otsu, 1979).

The method searches for a threshold that maximizes the separation between two intensity classes. In practical terms, the image histogram is divided into foreground and background groups, and the threshold that provides the strongest statistical separation between these classes is selected.

Otsu thresholding is particularly effective when the histogram has approximately two dominant intensity groups. It does not require a manually selected threshold and is therefore frequently used in document analysis, biomedical imaging, industrial inspection, and object-mask generation.

However, the method may perform poorly when the image contains substantial illumination variation, multiple object classes, strong noise, or overlapping intensity distributions. In such cases, adaptive thresholding, clustering, or learning-based segmentation may provide better results.

4.3. Edge Detection and Gradient Information

While thresholding separates regions according to intensity values, edge detection focuses on identifying rapid spatial changes in image intensity. These changes frequently correspond to object boundaries, surface transitions, or structural discontinuities.

The image gradient provides information about both the magnitude and direction of local intensity change. For a digital image, horizontal and vertical derivatives can be approximated using small convolution kernels.

Sobel filtering is one of the most widely known gradient-based techniques. It estimates intensity changes in horizontal and vertical directions using two kernels.

The gradient magnitude can be represented approximately as:

$$\mathbf{G = \sqrt{(Gx^2 + Gy^2)}}$$

where:

Gx = horizontal gradient response
Gy = vertical gradient response
G = overall gradient magnitude

For simpler implementation, the following approximation may also be used:

$$\mathbf{G \approx |Gx| + |Gy|}$$

Pixels with large gradient values are likely to correspond to image edges.

Sobel filtering is computationally efficient and also provides limited smoothing because of its kernel structure. However, it may produce multiple or noisy responses in complex images.

4.4. Canny Edge Detection

Canny edge detection provides a more structured approach to boundary extraction than simple gradient filtering (Canny, 1986). The method generally consists of several sequential operations:

Noise reduction → Gradient calculation → Non-maximum suppression → Double thresholding → Edge tracking

The initial smoothing stage reduces sensitivity to noise. Gradient information is then calculated to identify candidate edges. Non-maximum suppression removes responses that do not correspond to local gradient maxima, producing thinner boundaries.

Finally, two threshold values are used to distinguish strong and weak edges. Weak edges connected to strong boundaries may be retained, while isolated weak responses are removed.

The main advantage of the Canny method is its ability to generate relatively thin and continuous boundaries. However, its performance depends on parameters such as smoothing strength and threshold values.

Edge detection should therefore be regarded as a structural extraction method rather than a complete object-segmentation solution. Additional operations are often required to convert detected boundaries into meaningful regions.

4.5. Region-Based and Clustering-Based Segmentation

Thresholding and edge detection primarily use local intensity information. Region-based techniques instead aim to identify connected areas with similar visual characteristics.

Region growing begins from one or more initial pixels and progressively includes neighboring pixels that satisfy predefined similarity conditions. The effectiveness of this method depends on seed selection and similarity criteria.

Clustering methods provide another alternative. K-means clustering, for example, can group pixels according to intensity or color similarity without requiring predefined class labels.

Conceptually, the objective is to assign each pixel to the nearest cluster center. Although computationally straightforward, the method requires the number of clusters to be specified in advance and may be sensitive to initialization.

Clustering can be particularly useful when segmentation depends on multiple color channels or when clear threshold values cannot be identified from a single intensity histogram.

4.6. Mathematical Morphology

Segmentation results frequently contain small isolated regions, holes, disconnected structures, or irregular boundaries. Mathematical morphology provides a systematic framework for modifying such structures according to their geometric characteristics.

Morphological processing uses a small predefined pattern known as a **structuring element**. This element is moved across the image and determines how object boundaries are expanded, reduced, connected, or filtered.

Morphological techniques are applicable to both binary and grayscale images and remain important in image filtering, segmentation, shape analysis, and pattern recognition. Core operations include erosion, dilation, opening, and closing.

4.6.1. Erosion

Erosion reduces the boundaries of foreground objects. It can remove small isolated regions, separate objects connected by narrow structures, and reduce foreground thickness.

Conceptually:

Eroded Image = Original Image $\ominus$ Structuring Element

The symbol $\ominus$ represents morphological erosion.

The amount of object reduction depends on the size and shape of the selected structuring element.

4.6.2. Dilation

Dilation performs the opposite operation by expanding foreground regions.

Conceptually:

Dilated Image = Original Image $\oplus$ Structuring Element

The symbol $\oplus$ represents morphological dilation.

Dilation can connect nearby structures, enlarge object regions, and fill small discontinuities along boundaries.

4.6.3. Opening

Opening consists of erosion followed by dilation:

Opening = Erosion → Dilation

This operation is particularly useful for eliminating small foreground objects or narrow protrusions while approximately preserving the overall shape of larger structures.

4.6.4. Closing

Closing consists of dilation followed by erosion:

Closing = Dilation → Erosion

Closing is useful for filling small holes, connecting nearby foreground regions, and removing small gaps in object boundaries.

The relationships among thresholding, edge extraction, and morphological refinement are illustrated in Figure 4.

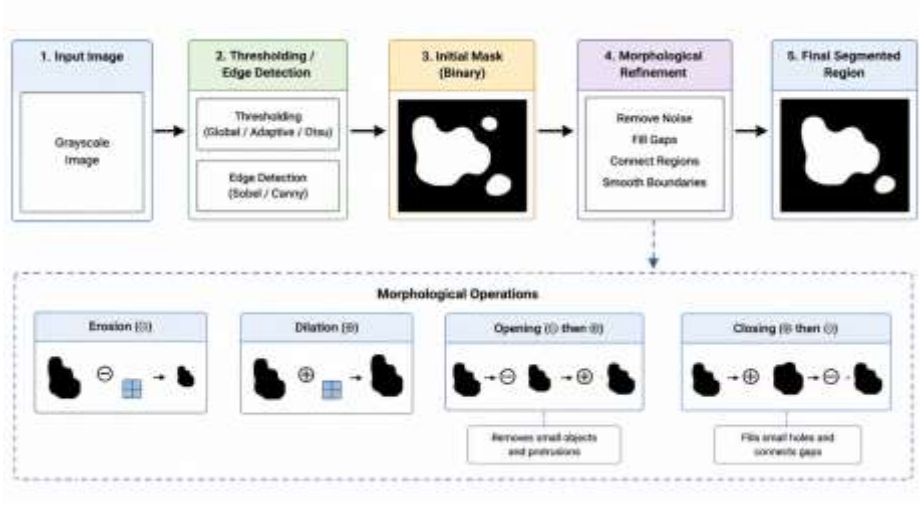


Figure 4. General workflow for structure extraction using thresholding, edge detection, and morphological refinement operations.

Figure 4 demonstrates that segmentation is often not performed using a single operation. An initial image may first be thresholded or processed using an edge detector. Morphological operations can then remove isolated responses,

connect discontinuous boundaries, and improve the structural consistency of the resulting mask.

4.7. Morphological Refinement of Segmentation Masks

Morphology is particularly important after threshold-based or learning-based segmentation because predicted masks may contain imperfections.

For example, a binary segmentation result may contain small false-positive regions. Opening can suppress these structures. Conversely, an object mask containing small internal gaps may benefit from closing.

The structuring element must be selected carefully. A large element may remove important small objects or excessively modify object boundaries, while an element that is too small may have little effect.

The most common structuring-element shapes include rectangular, elliptical, and cross-shaped patterns. Their suitability depends on the geometry of the structures being processed.

Morphological processing therefore provides a simple but powerful mechanism for introducing geometric prior knowledge into an image-processing pipeline.

4.8. From Classical to Learning-Based Segmentation

Classical segmentation methods rely primarily on explicitly defined image characteristics such as intensity, color, gradients, connectivity, or geometry. Their main advantages include low computational complexity, interpretability, and limited data requirements.

However, they may become unreliable when images contain complex backgrounds, overlapping objects, substantial appearance variation, weak boundaries, or non-uniform illumination.

Deep-learning-based segmentation addresses these challenges by learning feature representations from annotated examples. Convolutional neural networks have substantially advanced semantic and instance segmentation, while more recent transformer-based architectures use self-attention to model long-range relationships between image regions.

Transformer-based segmentation has become an important research direction because global contextual relationships can be particularly valuable in dense prediction tasks. Recent surveys show that transformer architectures are now widely investigated for semantic, instance, panoptic, medical, and other segmentation problems (Li et al., 2024).

Nevertheless, learning-based segmentation generally requires substantially greater computational resources and training data than classical approaches. Therefore, traditional methods remain valuable when data are limited, scene conditions are controlled, or computational simplicity is a priority.

The principal characteristics of the segmentation and structure-extraction methods discussed in this section are summarized in Table 3.

Table 3. Comparison of common segmentation and structure-extraction techniques.

Method	Main Principle	Main Advantage	Main Limitation
Global Thresholding	Single intensity threshold	Very simple and fast	Sensitive to illumination variation
Adaptive Thresholding	Local threshold calculation	Handles non-uniform illumination	Parameter dependent
Otsu Method	Automatic histogram-based threshold	No manual threshold required	Best for approximately bimodal distributions
Sobel	Gradient-based edge extraction	Computationally efficient	Sensitive to noise
Canny	Multi-stage edge detection	Thin and continuous edges	Requires parameter tuning
Region Growing	Neighbor similarity	Produces connected regions	Sensitive to seed selection

Method	Main Principle	Main Advantage	Main Limitation
K-means	Feature-based clustering	Simple unsupervised segmentation	Number of clusters must be defined
Morphology	Structuring-element operations	Effective mask refinement	Depends on structuring-element design
Deep Segmentation	Learned visual features	Strong performance in complex scenes	Requires data and computation

As summarized in Table 3, no segmentation method is universally superior. The appropriate technique depends on image complexity, computational constraints, available training data, and the type of structural information required.

In many practical systems, traditional and learning-based techniques can also be combined. A deep-learning model may generate an initial segmentation mask, while morphological opening or closing is subsequently applied to improve boundary continuity and remove isolated predictions. Similarly, threshold-based preprocessing may help isolate candidate regions before more computationally demanding analysis.

Modern segmentation should therefore be considered as a continuum ranging from simple pixel-level decisions to learned semantic interpretation. Despite rapid developments in CNN- and transformer-based approaches, classical thresholding, edge analysis, and morphology remain fundamental because they provide computationally efficient, interpretable, and easily controllable mechanisms for extracting structural information from digital images.

5. FROM CLASSICAL IMAGE PROCESSING TO LEARNING-BASED COMPUTER VISION

Classical image-processing methods rely primarily on explicitly defined mathematical operations and manually designed visual characteristics. Techniques such as filtering, thresholding, edge detection, morphology, and handcrafted feature extraction have been successfully used for decades in controlled or moderately complex environments. Their major advantages

include interpretability, relatively low computational cost, and limited dependence on large annotated datasets.

However, real-world visual data often contain substantial variability in illumination, background, object appearance, scale, orientation, and occlusion. Under such conditions, fixed processing rules may become insufficient. This limitation motivated the increasing use of machine-learning and deep-learning approaches capable of learning visual representations directly from data.

Modern computer vision therefore represents a transition from explicitly programmed visual rules toward data-driven representation learning. Rather than manually defining all relevant image characteristics, deep-learning models progressively learn features that are useful for the target task.

5.1. Handcrafted Features and Classical Machine Learning

Before the widespread adoption of deep learning, image-analysis systems commonly followed a multi-stage pipeline:

Image → Preprocessing → Feature Extraction → Feature Selection → Classifier → Decision

In this framework, the quality of the final prediction strongly depended on the design of the extracted features. Frequently used descriptors included edges, corners, texture statistics, shape features, local gradients, and keypoint-based representations.

Examples of classical descriptors include Histogram of Oriented Gradients (HOG), Scale-Invariant Feature Transform (SIFT), Local Binary Patterns (LBP), and Speeded-Up Robust Features (SURF) (Ojala et al., 2002; Lowe, 2004; Dalal & Triggs, 2005; Bay et al., 2008). These methods convert image content into numerical feature vectors that can subsequently be processed using classifiers such as support vector machines, k-nearest neighbors, decision trees, or random forests.

Classical machine-learning pipelines remain useful when datasets are small, computational resources are limited, or the relevant visual characteristics can be described using interpretable features. Nevertheless, feature engineering may require substantial domain expertise, and handcrafted descriptors may not generalize effectively to complex image variability.

5.2. Convolutional Neural Networks

Convolutional Neural Networks (CNNs) fundamentally changed computer vision by allowing visual features to be learned automatically from training data. Instead of defining edge, texture, or shape descriptors manually, convolutional layers learn suitable filters through optimization.

A simplified CNN processing sequence can be expressed as:

Input Image → Convolution → Activation → Feature Extraction → Classification / Detection / Segmentation

The convolution operation applies learnable kernels across local image regions. During training, kernel values are updated so that the resulting feature maps become increasingly informative for the target task.

Early CNN layers generally learn relatively simple patterns such as edges, orientation changes, or local textures. Deeper layers progressively combine these low-level patterns into more abstract representations such as object parts and semantic structures.

Recent surveys demonstrate that CNN-based architectures continue to play a central role in image classification, object detection, segmentation, super-resolution, medical imaging, remote sensing, and industrial inspection (Zhao et al., 2024).

5.3. Hierarchical Feature Learning

One of the main advantages of deep learning is hierarchical representation learning. Instead of relying on a single predefined descriptor, neural networks generate multiple levels of abstraction.

A conceptual representation is:

Pixels → Edges → Textures → Shapes → Object Parts → Semantic Features

This hierarchy enables deep models to adapt their internal representations according to the training objective.

For example, in object classification, deeper features may primarily represent category-specific information. In object detection, learned representations

must encode both semantic identity and spatial localization. In segmentation, feature representations must preserve sufficient spatial detail to generate pixel-level predictions.

This flexibility is one of the reasons why deep-learning approaches have achieved strong performance across a wide variety of computer-vision applications.

5.4. Transfer Learning

Training deep networks from the beginning may require large datasets and substantial computational resources. Transfer learning provides an alternative by reusing representations learned from a previously trained model.

A common transfer-learning workflow can be represented as:

Pretrained Model → Feature Reuse → Task-Specific Fine-Tuning → Target Application

Models pretrained on large-scale image datasets learn general-purpose visual characteristics that can often be transferred to smaller application-specific datasets.

Transfer learning is especially useful in medical imaging, industrial inspection, agriculture, remote sensing, and other domains where obtaining large labeled datasets may be difficult or expensive.

However, the effectiveness of transfer learning depends on the similarity between the source and target domains. If the visual characteristics are substantially different, pretrained representations may require extensive adaptation.

5.5. Attention Mechanisms

CNNs primarily capture local spatial patterns through convolution operations. Although deeper networks can model larger receptive fields, conventional convolution does not explicitly model all long-range relationships within an image.

Attention mechanisms address this limitation by allowing a model to assign different importance levels to different features or spatial regions.

Attention can conceptually be interpreted as:

Input Features → Importance Estimation → Feature Reweighting → Refined Representation

Channel attention emphasizes informative feature channels, while spatial attention focuses on important image regions.

Lightweight attention modules such as Squeeze-and-Excitation, Efficient Channel Attention, and combined channel-spatial attention mechanisms have been widely integrated into CNN architectures to improve feature discrimination with relatively limited computational overhead.

Attention mechanisms are particularly valuable when relevant image information occupies only a small portion of the input or when background structures may interfere with prediction.

5.6. Vision Transformers

The success of transformer architectures in natural language processing motivated their adaptation to computer vision. Vision Transformers (ViTs) process images by dividing them into smaller patches and modelling relationships among these patches using self-attention.

A simplified Vision Transformer workflow can be represented as:

Image → Image Patches → Patch Embeddings → Self-Attention → Feature Representation → Prediction

Unlike standard convolution, self-attention can directly model relationships between distant image regions. This provides a mechanism for capturing global contextual information.

Vision Transformers and their variants have been successfully applied to image classification, detection, segmentation, medical imaging, and remote sensing. However, traditional transformer architectures may require considerable computation and large training datasets. Consequently, efficient and hybrid transformer designs have become an important research direction (Papa et al., 2024).

5.7. Hybrid CNN–Transformer Architectures

CNNs and transformers provide different but complementary properties. CNNs are effective at learning local spatial structures and exhibit useful inductive biases for image processing. Transformers provide strong global-context modelling through attention.

Hybrid architectures attempt to combine these advantages.

A typical hybrid strategy may use convolutional layers for early feature extraction and transformer blocks for higher-level contextual modelling:

Input → CNN-Based Local Features → Transformer-Based Global Context → Prediction

Such designs are increasingly used when both fine local details and global relationships are important.

For example, segmentation of complex structures may require detailed boundaries together with an understanding of the overall scene. Similarly, object detection may benefit from local texture information and global contextual relationships.

5.8. Classical and Learning-Based Approaches as Complementary Methods

Deep learning has substantially improved the performance of computer-vision systems, but this does not eliminate the importance of traditional image processing.

In practice, classical preprocessing is frequently used before or after a neural network. Examples include normalization before training, contrast enhancement for low-quality images, morphology for refining segmentation masks, and filtering for suppressing acquisition noise.

Therefore, modern visual-analysis systems can often be represented using the following hybrid workflow:

Image Acquisition → Classical Preprocessing → Learned Feature Extraction → Prediction → Classical Post-processing

This combination may improve computational efficiency, interpretability, and robustness.

The transition from classical image processing toward modern learning-based visual analysis is illustrated in Figure 5.

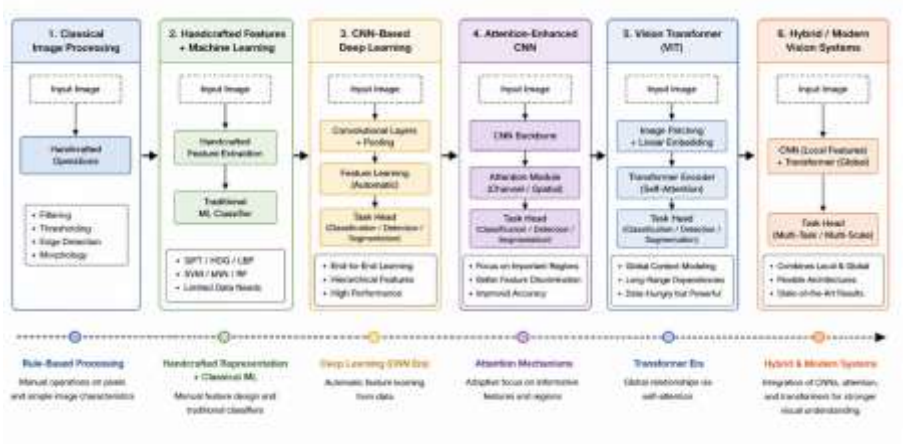


Figure 5. Evolution of computer-vision pipelines from handcrafted image-processing methods to deep-learning, attention-based, and transformer-based visual representation learning.

As illustrated in Figure 5, traditional systems rely heavily on manually defined processing rules and feature descriptors. Deep-learning systems replace much of this feature engineering with automatically learned representations, while attention and transformer-based models further extend the ability to capture long-range relationships and contextual information.

5.9. Comparison of Major Visual-Analysis Paradigms

Table 4 summarizes the major differences among classical image processing, traditional machine learning, CNN-based deep learning, and transformer-based computer vision.

Table 4. Comparison of major image-processing and computer-vision paradigms.

Approach	Feature Extraction	Data Requirement	Interpretability	Computational Cost	Main Strength
Classical Image Processing	Manually defined operations	Low	High	Low	Simple and controllable
Classical Machine Learning	Handcrafted descriptors	Low–Moderate	Moderate–High	Low–Moderate	Effective on structured data
CNN-Based Deep Learning	Automatically learned	Moderate–High	Moderate	Moderate–High	Strong local feature learning
Attention-Based CNN	Automatically learned with feature reweighting	Moderate–High	Moderate	Moderate	Improved feature discrimination
Vision Transformer	Patch-based learned representations	High	Moderate	High	Global context modelling
Hybrid CNN–Transformer	Local and global learned features	Moderate–High	Moderate	High	Combines complementary representations

As summarized in Table 4, classical methods generally provide greater interpretability and lower computational requirements, whereas learning-based approaches offer stronger adaptability for complex visual patterns.

No single paradigm is universally optimal. The appropriate strategy depends on factors such as dataset size, computational resources, latency requirements, interpretability needs, and image complexity.

5.10. Practical Considerations

The choice between classical and learning-based approaches should be guided by the requirements of the application rather than by the novelty of the method.

Classical methods may be preferable when:

- the imaging environment is controlled,
- visual characteristics are well defined,
- only limited training data are available,
- computational resources are constrained,
- or interpretability is a primary requirement.

Deep-learning methods may be more appropriate when:

- the visual data contain substantial variability,
- relevant features are difficult to define manually,
- sufficient labeled data are available,
- and higher computational complexity is acceptable.

Hybrid approaches may provide an effective compromise when domain knowledge and learned representations can be integrated within the same pipeline.

5.11. Current Direction of Modern Computer Vision

Current computer-vision research increasingly focuses on efficient architectures, multimodal representations, foundation models, self-supervised learning, and general-purpose visual understanding.

One important trend is the development of models that can perform multiple visual tasks rather than being trained for only one application. Another is the

use of large-scale pretraining to produce representations that can be adapted to downstream tasks using relatively limited application-specific data.

At the same time, computational efficiency has become increasingly important. Models intended for edge devices, embedded systems, mobile platforms, or real-time applications must provide a balance between accuracy, parameter count, memory usage, and inference speed.

Consequently, the direction of modern computer vision is not simply toward increasingly large models. Instead, current research emphasizes the development of visual systems that are accurate, efficient, adaptable, and capable of integrating both local and global information.

Overall, the transition from classical image processing to learning-based computer vision reflects a broader evolution from manually specified representations toward automatically learned visual knowledge. Nevertheless, conventional image-processing techniques remain essential components of many practical systems. The strongest contemporary pipelines frequently combine classical preprocessing, learned feature extraction, attention-based representation refinement, and task-specific post-processing within a unified framework.

6. APPLICATIONS, CHALLENGES, AND FUTURE DIRECTIONS

Image processing and computer vision have evolved from relatively simple pixel-level operations into intelligent visual-analysis systems capable of supporting complex decision-making processes. Today, these technologies are used in healthcare, autonomous systems, industrial inspection, agriculture, remote sensing, security, robotics, infrastructure monitoring, and many other application domains.

The practical value of image-processing techniques depends not only on algorithmic accuracy but also on factors such as computational efficiency, robustness, interpretability, data availability, and deployment conditions. Consequently, contemporary computer-vision research increasingly focuses on developing systems that provide reliable performance under realistic operating environments rather than only under controlled laboratory conditions.

6.1. Medical and Healthcare Applications

Medical imaging represents one of the most important application areas of image processing and computer vision. Digital images obtained from magnetic resonance imaging, computed tomography, ultrasound, X-ray, microscopy, and endoscopic systems may contain large amounts of clinically relevant information.

Conventional image-processing methods are frequently used for contrast enhancement, denoising, boundary extraction, image registration, and segmentation. Deep-learning models extend these capabilities by automatically learning representations suitable for classification, lesion detection, anatomical segmentation, and decision-support applications.

A common medical image-analysis pipeline can be represented as:

Medical Image → Preprocessing → Region Analysis → Feature Learning → Prediction / Decision Support

Image quality remains especially important in medical applications because subtle structures may carry diagnostic significance. Excessive filtering, unsuitable resizing, or aggressive enhancement can remove relevant information. Therefore, preprocessing parameters should be selected carefully according to the imaging modality and target task.

Edge-based deep-learning systems are also receiving increasing attention because they can support lower-latency and privacy-sensitive medical applications without requiring all visual data to be transferred continuously to remote servers. Recent reviews identify efficient edge deployment as an important direction for both general computer vision and medical diagnostics (Mittal, 2024).

6.2. Industrial Inspection and Quality Control

Automated visual inspection has become an important component of modern manufacturing systems. Cameras and imaging sensors can continuously inspect products, surfaces, components, and production processes without requiring direct human observation of every item.

Typical industrial computer-vision tasks include:

- surface-defect detection,

- dimensional inspection,
- component verification,
- anomaly detection,
- product classification,
- assembly monitoring,
- and quality assessment.

Traditional image-processing methods remain effective when defect characteristics are well defined. For example, thresholding and morphology may be sufficient for identifying simple shape irregularities or missing components under controlled illumination.

More complex defects, however, may involve irregular textures, subtle surface patterns, and substantial appearance variability. Deep-learning-based detectors and segmentation models are therefore increasingly used for automated inspection.

A major requirement in industrial systems is real-time operation. Models must therefore provide an appropriate balance between accuracy and computational complexity. Beyond conventional surface inspection, computer-vision approaches can also be extended to non-destructive subsurface sensing. Arvas and Çınar (2026) demonstrated that a lightweight YOLO-based approach can be used for detecting underground objects from Ground Penetrating Radar (GPR) data, illustrating the potential of efficient deep-learning models for subsurface inspection and non-destructive evaluation applications.

6.3. Autonomous Systems and Intelligent Transportation

Modern transportation systems rely extensively on visual information. Cameras mounted on vehicles, roads, unmanned aerial vehicles, and traffic infrastructure can provide continuous information about the surrounding environment.

Major applications include:

Vehicle Detection → Pedestrian Detection → Traffic-Sign Recognition → Lane Analysis → Scene Understanding → Navigation

Object detection and semantic segmentation are particularly important because autonomous systems must identify both objects and the spatial organization of the environment.

Visual-analysis models deployed in vehicles or embedded platforms must generally operate under strict latency and energy constraints. Environmental variations such as rain, fog, nighttime illumination, shadows, motion blur, and occlusion further complicate the problem.

For this reason, lightweight computer-vision architectures have become an important research direction. Recent surveys emphasize that edge-oriented detection systems must achieve a balance among model size, latency, memory consumption, and recognition accuracy (Mittal, 2024).

6.4. Agriculture and Environmental Monitoring

Image-processing techniques are increasingly used in precision agriculture and environmental monitoring. Images acquired from ground cameras, unmanned aerial vehicles, satellites, and robotic platforms can provide information over large areas without requiring continuous manual inspection.

Representative applications include:

- crop and plant identification,
- disease detection,
- fruit detection and counting,
- weed recognition,
- vegetation monitoring,
- environmental change analysis,
- and land-cover classification.

Color information, texture, shape, and spectral characteristics can be analyzed using conventional image-processing methods. However, environmental conditions introduce considerable variability in illumination, background structure, plant appearance, scale, and occlusion.

Deep-learning models can provide greater adaptability under these conditions, while classical preprocessing remains useful for normalization, illumination correction, color-space transformation, and image enhancement.

6.5. Remote Sensing and Earth Observation

Remote-sensing images provide large-scale information about the Earth's surface. Satellite and aerial imagery can be analyzed for urban planning, environmental assessment, agriculture, disaster management, infrastructure monitoring, and geographical mapping.

Remote-sensing images may contain multiple spectral channels and extremely high spatial resolution. As a result, computer-vision systems in this field must process large image dimensions while preserving small spatial structures.

Common tasks include:

Classification → Object Detection → Semantic Segmentation → Change Detection

Modern CNN and transformer architectures are increasingly applied to these problems because they can learn spatial and contextual relationships across complex scenes.

However, remote-sensing applications also present challenges such as domain variation, limited annotations, class imbalance, small-object representation, and substantial computational requirements.

6.6. Security, Surveillance, and Robotics

Computer vision also plays an important role in surveillance and robotic perception. Surveillance systems may use visual-analysis methods for object detection, tracking, anomaly recognition, and activity analysis.

Robotic systems require visual perception to interact with dynamic environments. Tasks may include object localization, obstacle detection, pose estimation, navigation, grasping, and scene understanding.

These applications frequently require real-time inference. Consequently, computational efficiency becomes as important as predictive performance.

The increasing integration of visual models into embedded devices has accelerated research on model compression, pruning, quantization, knowledge distillation, and lightweight architecture design. Efficient deep-learning methods are particularly important for autonomous vehicles, robotics, mobile devices, and edge-computing systems because large visual models may otherwise introduce excessive latency, energy consumption, and memory demands.

6.7. Major Challenges in Modern Computer Vision

Despite substantial advances, computer-vision systems still face several important limitations.

6.7.1. Data Dependency

Supervised deep-learning models generally require substantial quantities of representative labeled data. Preparing such datasets can be expensive and time-consuming, particularly for medical, industrial, and scientific applications requiring expert annotation.

Dataset imbalance can create an additional problem. If some classes contain substantially more samples than others, a model may become biased toward frequently observed categories.

Transfer learning, self-supervised learning, synthetic data generation, and few-shot learning are increasingly investigated to reduce dependence on large task-specific labeled datasets.

6.7.2. Domain Shift and Generalization

A model that performs well on training data may produce weaker results when environmental conditions change.

Differences may arise from:

Camera → Illumination → Background → Sensor → Location → Image Resolution

This phenomenon is commonly referred to as domain shift.

Developing models that generalize across different acquisition environments remains one of the major challenges in computer vision. Robust

preprocessing, data augmentation, domain adaptation, and large-scale pretraining can reduce this problem but cannot eliminate it completely.

6.7.3. Computational Complexity

Increasing model size and architectural complexity may improve representation capability but also increases memory usage, energy consumption, and inference latency.

These requirements are problematic for real-time or embedded applications.

A useful practical system must therefore balance:

Accuracy ↔ Speed ↔ Memory ↔ Energy Consumption

Model compression, quantization, pruning, knowledge distillation, and efficient architecture design represent important strategies for improving this balance. Current literature increasingly emphasizes computation-efficient computer vision because highly accurate models are of limited practical value when their latency or power requirements prevent real-world deployment.

6.7.4. Explainability and Reliability

Deep-learning models may produce highly accurate predictions while providing limited information regarding the reasoning underlying those predictions.

This becomes particularly important in high-stakes domains such as healthcare, autonomous systems, and critical infrastructure.

Explainable artificial intelligence methods attempt to identify the image regions or features that contribute most strongly to a model prediction. Examples include activation maps, feature visualization, attribution methods, and attention analysis.

However, explainability should not be confused automatically with reliability. A visually plausible explanation does not necessarily prove that the model has learned an appropriate causal relationship.

The emergence of vision foundation models further increases the importance of interpretability because such models may contain extremely large and complex learned representations. Recent work therefore identifies

explainability, evaluation, bias, adversarial vulnerability, and contextual understanding as major open problems for foundation-model-based vision systems (Awais et al., 2025).

6.8. Vision Foundation Models and Multimodal Systems

One of the most significant recent developments in computer vision is the emergence of foundation models trained on large and diverse datasets.

Traditional supervised computer-vision systems are generally trained for a specific task. In contrast, foundation models aim to learn general visual representations that can subsequently be adapted to different applications (Awais et al., 2025).

A conceptual transition can be expressed as:

Task-Specific Model → Large-Scale Pretraining → General Visual Representation → Multiple Downstream Tasks

Vision foundation models may support classification, detection, segmentation, retrieval, visual question answering, and other tasks using a common pretrained representation.

Another important development is the integration of vision with language. Vision-language models can associate images with textual concepts, enabling more flexible interaction and semantic interpretation (Zhang et al., 2024).

Recent surveys show that multimodal foundation models increasingly replace conventional single-task pretraining strategies and can be adapted to downstream applications through few-shot learning, prompt-based interaction, or parameter-efficient fine-tuning.

Foundation models also introduce new possibilities for prompt-driven computer vision. Instead of retraining an entire model for every task, users may specify desired objects or visual concepts through textual or visual prompts. A recent IEEE TPAMI survey identifies multimodal reasoning, prompting, transferability, and general-purpose visual understanding as central characteristics of this emerging paradigm.

6.9. Multimodal Visual Analysis

Many real-world applications contain information beyond visible RGB imagery.

Available modalities may include:

RGB + Depth + Thermal + Radar + LiDAR + Text + Audio

Combining multiple information sources can improve robustness when one modality becomes unreliable.

For example, visible cameras may perform poorly under low illumination, while thermal imagery can retain useful object information. Similarly, depth sensors can provide geometric information that is difficult to infer reliably from RGB images alone.

Multimodal fusion can be performed at several stages, including input-level fusion, feature-level fusion, and decision-level fusion.

Recent literature indicates growing interest in hybrid multimodal architectures that combine complementary representations from different sensors or information sources. Such methods are particularly relevant to autonomous systems, healthcare, robotics, and remote sensing (Zhang et al., 2024).

6.10. Edge Intelligence and Efficient Vision

The increasing deployment of computer vision outside cloud environments has created strong demand for models that can operate on embedded hardware.

Edge intelligence refers to performing artificial-intelligence operations near the location where data are generated.

A typical edge-vision pipeline may be represented as:

Sensor → On-Device Preprocessing → Lightweight Model → Local Decision

This architecture can reduce communication latency and may provide advantages in privacy, reliability, and bandwidth usage.

However, edge devices generally have limited computational resources. Efficient model design therefore requires careful consideration of parameter

count, floating-point operations, memory access, energy consumption, and inference latency.

Recent surveys of lightweight object detection and edge deep learning identify architecture design, model compression, quantization, pruning, and hardware-aware optimization as major strategies for enabling practical computer vision on resource-constrained devices (Mittal, 2024).

6.11. Future Directions

The future of computer vision is likely to involve greater integration among classical image processing, deep learning, multimodal information, and foundation models.

Major research directions can be summarized as:

Efficient Vision → Foundation Models → Multimodal Learning → Self-Supervised Learning → Explainable AI → Edge Intelligence

These directions are conceptually illustrated in Figure 6.

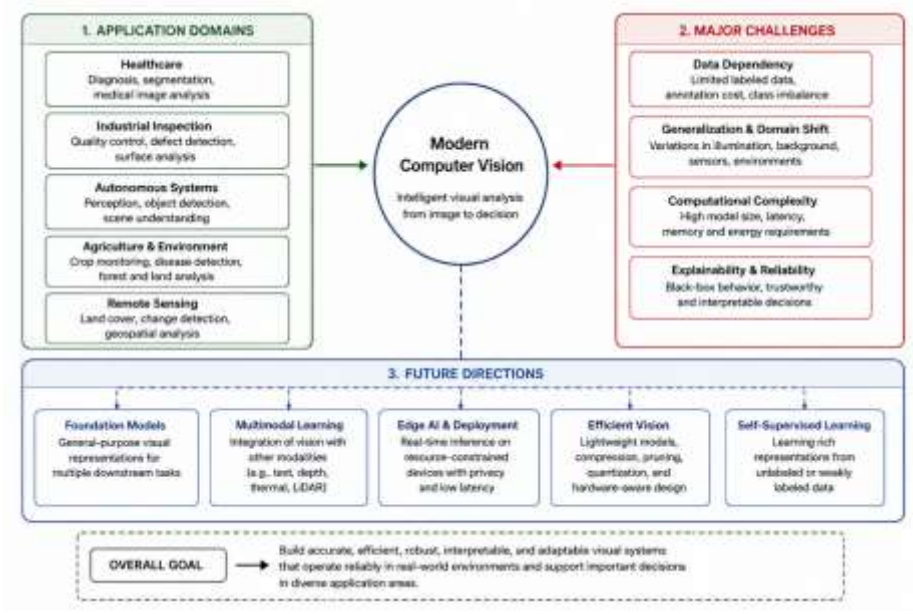


Figure 6. Major application domains, current challenges, and emerging research directions in modern image processing and computer vision.

Figure 6 should be interpreted as an interconnected framework rather than a linear technological sequence. Future visual systems are expected to combine multiple capabilities. For example, a multimodal foundation model may require efficient edge deployment, while an autonomous system may simultaneously require real-time inference, explainability, robustness, and multimodal sensor fusion.

The main characteristics of these emerging research directions are summarized in Table 5.

Table 5. Major emerging directions in modern image processing and computer vision.

Research Direction	**Main Objective**	**Principal Advantage**	**Major Challenge**
Lightweight Vision Models	Reduce computational complexity	Real-time and edge deployment	Accuracy-efficiency trade-off
Self-Supervised Learning	Learn from unlabeled data	Reduced annotation requirements	Reliable representation design
Vision Transformers	Model global dependencies	Strong contextual representation	Computational cost
Foundation Models	General-purpose visual representations	Adaptation to multiple tasks	Scale and reliability
Vision-Language Models	Integrate visual and textual information	Flexible semantic interaction	Multimodal alignment
Explainable Vision	Interpret model decisions	Improved transparency	Faithful explanations

Research Direction	Main Objective	Principal Advantage	Major Challenge
Edge Intelligence	Perform inference near data source	Low latency and privacy	Hardware constraints
Multimodal Fusion	Combine complementary sensors	Increased robustness	Data synchronization and fusion

As summarized in Table 5, future research is increasingly concerned with making visual systems not only more accurate but also more adaptable, efficient, interpretable, and generalizable.

The rapid development of foundation models illustrates this transition clearly. Instead of designing a completely independent model for every visual task, future systems may increasingly rely on large pretrained representations that can be adapted using limited examples, prompts, or lightweight fine-tuning. At the same time, practical deployment will require reducing the computational demands associated with such models.

Therefore, the next generation of computer-vision systems will likely emerge from the interaction of two apparently conflicting trends: increasingly powerful general-purpose models and increasingly efficient deployment strategies. The ability to combine these objectives while maintaining reliability, interpretability, and robustness will remain a major research challenge.

7. CONCLUSION

Digital image processing has evolved from fundamental pixel-level operations into a broad computational framework that supports intelligent visual analysis and automated decision-making. Despite the rapid development of deep learning, attention mechanisms, and transformer-based architectures, the fundamental principles of digital image representation, preprocessing, enhancement, segmentation, edge analysis, and mathematical morphology remain essential for understanding and designing reliable computer-vision systems.

This chapter presented image processing as a continuous workflow extending from image acquisition and digital representation to intelligent visual interpretation. Initially, fundamental concepts including pixels, sampling, quantization, bit depth, grayscale representation, binary images, and color spaces were discussed. These concepts establish the numerical foundation on which both conventional image-processing algorithms and modern learning-based approaches operate.

Preprocessing and enhancement techniques were subsequently examined as important mechanisms for improving the quality and consistency of visual information. Intensity normalization, gamma correction, histogram-based enhancement, noise filtering, sharpening, and resizing can significantly influence the performance of subsequent analytical stages. However, preprocessing should be selected according to image characteristics and application requirements because excessive or inappropriate transformations may remove useful information or introduce unwanted artifacts.

Segmentation, edge detection, and mathematical morphology provide mechanisms for converting pixel-level information into meaningful structural representations. Threshold-based techniques, Otsu's method, gradient-based edge detection, Canny processing, clustering, and morphological operations continue to provide computationally efficient and interpretable solutions for many applications. Although learning-based segmentation models have substantially improved performance in complex visual environments, conventional methods remain particularly useful when datasets are limited, scene characteristics are controlled, or transparent processing is required.

The transition from handcrafted image analysis to learning-based computer vision represents one of the most significant developments in the field. Traditional approaches rely on manually defined descriptors and processing rules, whereas convolutional neural networks automatically learn hierarchical visual features from data. Attention mechanisms further improve feature discrimination, while Vision Transformers extend visual modelling by capturing long-range contextual relationships. Hybrid architectures increasingly combine convolutional and transformer-based representations to exploit both local and global visual information.

Importantly, classical image processing and modern artificial intelligence should not be regarded as mutually exclusive alternatives. In practical systems, preprocessing, learned representation extraction, prediction, and

conventional post-processing are frequently integrated within the same computational pipeline. Such hybrid approaches can provide advantages in terms of robustness, computational efficiency, interpretability, and application-specific control.

The applications discussed in this chapter demonstrate the broad impact of image processing and computer vision. Healthcare, industrial inspection, autonomous systems, agriculture, remote sensing, security, robotics, and environmental monitoring increasingly rely on automated visual-analysis technologies. However, real-world deployment introduces challenges that cannot be evaluated solely through predictive accuracy. Data availability, domain shift, computational requirements, latency, energy consumption, explainability, and reliability must also be considered.

Emerging research directions indicate that future computer-vision systems will increasingly rely on foundation models, multimodal learning, self-supervised representation learning, efficient architectures, and edge intelligence. Large-scale pretrained visual models provide greater adaptability across multiple tasks, while multimodal systems allow complementary information from images, text, depth, thermal sensors, radar, LiDAR, and other data sources to be integrated. At the same time, the growing demand for real-time and embedded applications requires increasingly efficient model designs.

The future of image processing is therefore unlikely to be defined by a single algorithmic family. Instead, effective visual systems will combine fundamental signal and image-processing principles with data-driven representation learning. Classical methods can provide efficient and interpretable transformations, while modern learning architectures contribute adaptability and semantic understanding. The integration of these complementary capabilities offers a practical pathway toward visual-analysis systems that are accurate, efficient, robust, interpretable, and suitable for deployment under real-world conditions.

Overall, understanding the progression from digital image formation to intelligent visual interpretation provides a comprehensive foundation for contemporary computer vision. Although the technologies used to analyze images continue to evolve rapidly, the fundamental objective remains unchanged: to transform raw visual data into meaningful and reliable

information that can support scientific analysis, automated systems, and informed decision-making.

REFERENCES

Arvas, M. M., & Çınar, A. (2026). Detection of underground objects from GPR data using a lightweight YOLO-based approach. *Scientific Reports*.

Awais, M., Naseer, M., Khan, S., Anwer, R. M., Cholakkal, H., Shah, M., Yang, M.-H., & Khan, F. S. (2025). Foundation models defining a new era in vision: A survey and outlook. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 47*(4), 2245–2264. https://doi.org/10.1109/TPAMI.2024.3506283.

Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008). Speeded-Up Robust Features (SURF). *Computer Vision and Image Understanding, 110*(3), 346–359. https://doi.org/10.1016/j.cviu.2007.09.014.

Brar, K. K., Goyal, B., Dogra, A., Mustafa, M. A., Majumdar, R., Alkhayyat, A., & Kukreja, V. (2025). Image segmentation review: Theoretical background and recent advances. *Information Fusion, 114*, 102608. https://doi.org/10.1016/j.inffus.2024.102608.

Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-8*(6), 679–698.

Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. *Proceedings of CVPR*, 886–893.

Lepcha, D. C., Goyal, B., Dogra, A., Sharma, K. P., & Gupta, D. N. (2023). A deep journey into image enhancement: A survey of current and emerging trends. *Information Fusion, 93*, 36–76. https://doi.org/10.1016/j.inffus.2022.12.012.

Li, X., Ding, H., Yuan, H., Zhang, W., Pang, J., Cheng, G., Chen, K., Liu, Z., & Loy, C. C. (2024). Transformer-based visual segmentation: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 46*(12), 10138–10163. https://doi.org/10.1109/TPAMI.2024.3434373.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision, 60*, 91–110.

Mao, J., Sun, L., Chen, J., & Yu, S. (2025). Overview of research on digital image denoising methods. *Sensors, 25*(8), 2615. https://doi.org/10.3390/s25082615.

Mittal, P. (2024). A comprehensive survey of deep learning-based lightweight object detection models for edge devices. *Artificial Intelligence Review, 57*, 242. https://doi.org/10.1007/s10462-024-10877-1.

Münke, F. R., Schützke, J., Berens, F., & Reischl, M. (2024). A review of adaptable conventional image processing pipelines and deep learning on limited datasets. *Machine Vision and Applications, 35*, 25. https://doi.org/10.1007/s00138-023-01501-3.

Ojala, T., Pietikäinen, M., & Mäenpää, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 24*(7), 971–987.

Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics, 9*(1), 62–66. https://doi.org/10.1109/TSMC.1979.4310076.

Papa, L., Russo, P., Amerini, I., & Zhou, L. (2024). A survey on efficient vision transformers: Algorithms, techniques, and performance benchmarking. *IEEE*

Transactions on Pattern Analysis and Machine Intelligence, 46(12), 7682–7700. https://doi.org/10.1109/TPAMI.2024.3392941.
Szeliski, R. (2022). *Computer Vision: Algorithms and Applications* (2nd ed.). Springer. https://doi.org/10.1007/978-3-030-34372-9.
Zangana, H. M., Mohammed, A. K., Sallow, Z. B., & Mustafa, F. M. (2024). Exploring image representation and color spaces in computer vision: A comprehensive review. *Indonesian Journal of Computer Science, 13*(3), 4261–4272. https://doi.org/10.33022/ijcs.v13i3.3998.
Zhang, J., Huang, J., Jin, S., & Lu, S. (2024). Vision-language models for vision tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 46*(8), 5625–5644. https://doi.org/10.1109/TPAMI.2024.3369699.
Zhao, X., Wang, L., Zhang, Y., Han, X., Deveci, M., & Parmar, M. (2024). A review of convolutional neural networks in computer vision. *Artificial Intelligence Review, 57*, 99. https://doi.org/10.1007/s10462-024-10721-6.

Human–Computer Interaction in the Era of Intelligent and Emerging Technologies

Muhammed Mücahit ARVAS

Yazılım Mühendisliği Bölümü, Iğdır Üniversitesi, Iğdır, Türkiye E-mail: mmucahit.arvas@igdir.edu.tr

1. INTRODUCTION

Human–Computer Interaction (HCI) is a multidisciplinary research field concerned with understanding, designing, and evaluating the interactions that emerge between people and computational systems. Although the discipline initially developed around improving the usability of conventional computer interfaces, its scope has progressively expanded as computing technologies have become embedded in everyday environments, professional activities, communication systems, and decision-making processes. Contemporary HCI therefore extends beyond the design of screens, menus, and input devices; it investigates the broader relationship between human capabilities, technological behavior, contextual factors, and user experience.

The evolution of HCI has closely followed major transformations in computing. Early computer systems primarily relied on command-line interaction and required users to possess considerable technical knowledge. The emergence of graphical user interfaces subsequently transformed computers from specialist tools into technologies that could be used by much broader populations. Concepts such as visibility, feedback, consistency, mapping, learnability, and error prevention consequently became central considerations in interaction design (Norman, 2013; Shneiderman et al., 2016). The rapid development of the Web introduced another major transition by shifting interaction from isolated desktop applications toward distributed information environments. This transformation was followed by mobile computing, touch-based interfaces, wearable devices, ubiquitous computing, and increasingly context-sensitive digital services.

These technological transitions have gradually altered the conceptual position of the user within interactive systems. In traditional command-based environments, users were primarily responsible for explicitly specifying what the computer should perform. Graphical and web-based interfaces reduced this burden through visual representations and structured navigation. Mobile and ubiquitous systems subsequently introduced context awareness, allowing computational systems to respond to information such as location, movement, device state, and patterns of use. More recently, artificial intelligence (AI) has produced a substantially different interaction paradigm in which systems can infer intentions, generate content, recommend actions, and adapt their behavior according to user input and contextual information.

This transformation has contributed to the emergence of Human-Centered Artificial Intelligence (HCAI) as an important research direction situated at the intersection of HCI and AI. Rather than evaluating intelligent systems exclusively according to computational performance, HCAI emphasizes human needs, capabilities, values, agency, and societal consequences throughout the development and use of AI technologies. Capel and Brereton (2023) demonstrate that contemporary HCAI research increasingly connects interaction design with ethical AI, while also emphasizing the need for stronger integration between the HCI and AI research communities. In this perspective, successful intelligent systems should not merely achieve high predictive accuracy; they should also support meaningful human control, appropriate understanding, usability, accessibility, and responsible decision making.

The growing adoption of generative artificial intelligence has further accelerated this transformation. Large language models and multimodal generative systems enable users to communicate with computational systems through natural language, images, speech, and combinations of multiple modalities. Consequently, interaction is increasingly moving from predefined commands and menu structures toward conversational and collaborative forms of engagement. Users may now ask systems to summarize information, generate software code, produce visual material, support creative activities, analyze documents, or assist decision-making through comparatively open-ended instructions. Such systems therefore challenge traditional assumptions regarding how interfaces should be structured and how users should understand system behavior.

Generative AI also introduces new cognitive demands. Unlike conventional deterministic interfaces, the output produced by generative systems may vary between interactions and may contain incomplete, inappropriate, or inaccurate information. Users must therefore determine how a request should be formulated, evaluate the credibility of generated results, decide when additional verification is required, and judge how much reliance should be placed on system recommendations. Tankelevitch et al. (2024) describe these processes in terms of increased metacognitive demands, emphasizing that users must continuously monitor both their own interaction strategies and the reliability of AI-generated outputs. From an HCI perspective, this creates a need for interfaces that facilitate evaluation, revision, transparency, and

appropriate user control rather than encouraging passive dependence on automation.

A related change concerns the transition from explainable AI toward interactive AI. Traditional research on explainability has often concentrated on producing explanations for algorithmic predictions. However, effective human–AI interaction increasingly requires a broader perspective that considers how humans can interrogate, correct, refine, contest, or collaborate with intelligent systems over time. Raees et al. (2024) argue that contemporary human–AI research must address the relationship between system autonomy and user agency rather than treating explanation as an isolated interface feature. Accordingly, trustworthy interaction depends not only on presenting information about an AI decision but also on allowing users to meaningfully influence the interaction process.

The historical transformation summarized in Figure 1 demonstrates that HCI has evolved from explicit and predominantly device-oriented interaction toward increasingly natural, contextual, immersive, and intelligent forms of engagement. Each technological transition has also introduced a different set of design requirements. Command-line interfaces emphasized accuracy and technical knowledge, whereas graphical interfaces increased the importance of learnability and visual consistency. Web and mobile environments introduced information architecture, responsiveness, and contextual use. Immersive environments created new requirements associated with presence, spatial interaction, and physical comfort. Intelligent and generative systems now add challenges related to trust, transparency, user agency, uncertainty, and human–AI collaboration.

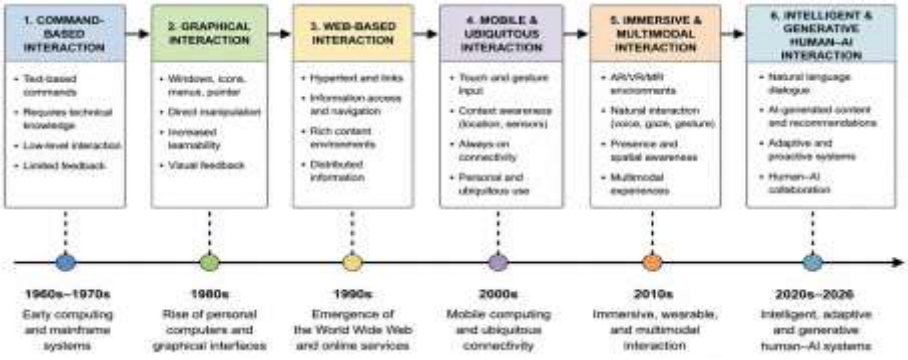


Figure 1. Evolution of human–computer interaction paradigms from command-based interfaces to intelligent, adaptive, and collaborative systems.

Contemporary HCI is also increasingly multimodal. Interaction may involve combinations of touch, speech, gaze, gesture, body movement, physiological signals, and natural-language communication. Augmented reality (AR), virtual reality (VR), mixed reality, wearable computing, and emerging spatial-computing platforms further blur the boundary between physical and digital environments. As interaction becomes distributed across devices and sensory modalities, designers must consider not only whether a system is technically functional but also how effectively it integrates with human perception, cognition, physical behavior, and social context.

Accessibility and inclusion represent another fundamental dimension of this transformation. Interactive systems are increasingly used by heterogeneous populations differing in age, physical abilities, cognitive characteristics, cultural backgrounds, digital literacy, and situational constraints. Human-centered design must therefore account for diversity from the earliest stages of development rather than treating accessibility as a later modification. Inclusive interaction benefits not only users with permanent disabilities but also individuals experiencing temporary or situational limitations, such as operating a device with one hand, interacting in a noisy environment, or using technology under high cognitive workload.

The growing influence of data-driven and intelligent systems has simultaneously increased the importance of privacy, ethics, fairness, and responsible design. An interface may be highly efficient while still influencing users through opaque recommendations, excessive data collection, deceptive interaction patterns, or algorithmically biased outcomes. Consequently, contemporary HCI evaluation cannot be restricted to conventional measures such as task completion time, error rate, or satisfaction. Designers and researchers increasingly need to consider whether systems preserve meaningful user choice, communicate uncertainty appropriately, protect personal information, support equitable access, and avoid manipulative interaction practices.

These developments indicate a broader conceptual transition in HCI: the objective is no longer simply to make computers easier to operate. The contemporary challenge is to establish interaction ecosystems in which humans can understand, influence, collaborate with, and appropriately trust increasingly autonomous computational systems. Human factors, user experience, usability, accessibility, cognition, ethics, and technological

intelligence therefore need to be examined as interconnected components rather than independent design concerns.

Within this context, this chapter examines the transition of HCI from conventional user-interface principles toward intelligent, adaptive, immersive, and human-centered interaction paradigms. First, the fundamental principles of human cognition, interaction design, and established HCI models are discussed. Human-centered design and contemporary usability evaluation are then examined, followed by emerging forms of interaction involving artificial intelligence, generative AI, multimodal systems, and immersive technologies. Particular attention is given to trust, transparency, accessibility, privacy, ethics, and appropriate human control. Finally, emerging research directions are discussed to provide a forward-looking perspective on the future relationship between humans and computational technologies.

2. FOUNDATIONS OF HUMAN–COMPUTER INTERACTION

Human–Computer Interaction is fundamentally concerned with the relationship between human capabilities and the behavior of interactive technologies. Effective interaction therefore requires more than functional software or visually attractive interfaces. Designers must understand how users perceive information, allocate attention, remember previous actions, construct mental models, make decisions, perform physical actions, and interpret system feedback. These human characteristics establish practical boundaries within which interactive systems should operate.

Contemporary HCI increasingly treats interaction as a dynamic process rather than a simple sequence of commands and responses. In this process, the user develops an intention, selects an action, interacts with an interface, observes the resulting system state, evaluates whether the outcome corresponds to the original goal, and modifies subsequent behavior when necessary. Modern intelligent systems extend this cycle by introducing prediction, adaptation, recommendation, and autonomous decision support. Consequently, contemporary interaction design must account for both traditional usability principles and emerging issues such as uncertainty, cognitive workload, personalization, and appropriate levels of automation.

2.1. Human Factors in Interactive Systems

Human factors represent the physical, perceptual, cognitive, and behavioral characteristics that influence interaction with technological systems. Designing without considering these characteristics can increase task duration, error probability, fatigue, frustration, and inappropriate system use. Conversely, interfaces that correspond to natural human capabilities can reduce unnecessary effort and improve both usability and user experience.

Three human characteristics are particularly important in interface design: perception, attention, and memory. Perception determines how users identify and interpret visual, auditory, or tactile information. Attention determines which elements receive cognitive priority at a particular moment, whereas memory influences the extent to which information from previous interactions can be retained and reused.

Visual organization is therefore a fundamental HCI consideration. Users should be able to distinguish important elements, understand relationships among interface components, and identify available actions without excessive visual search. Appropriate hierarchy, grouping, spacing, contrast, and consistency support this process. Interfaces containing excessive information, weak hierarchy, or competing visual elements may force users to spend cognitive resources determining where to focus rather than completing the intended task.

Attention is similarly limited. Digital environments increasingly compete for user attention through notifications, dynamic content, animations, alerts, and multiple simultaneously active applications. While such mechanisms may provide useful information, poorly prioritized signals can interrupt task flow and increase cognitive demands. This issue has become particularly important as computing has shifted from isolated desktop environments toward mobile, wearable, immersive, and continuously connected systems.

Memory constraints also affect interaction design. Interfaces should generally favor recognition over recall, allowing users to identify relevant actions from visible alternatives instead of requiring them to remember commands or procedures. Consistent navigation structures, meaningful labels, recently used items, contextual suggestions, and appropriate defaults can reduce memory demands. These principles remain relevant even as AI-enabled systems increasingly predict user intentions.

2.2. Mental Models and Interaction Expectations

A mental model refers to the internal representation that users develop regarding how a system operates. Users rarely understand the complete technical architecture of an application; instead, they construct simplified explanations based on previous experience, interface cues, observed system responses, and familiar real-world concepts.

When the conceptual model presented by a system corresponds reasonably well with the user's mental model, interaction becomes easier to predict and learn. When these models conflict, users may make errors even when the underlying system functions correctly. For example, if an interface uses familiar visual symbols but assigns unexpected behaviors to them, users must suppress previously learned expectations and develop a new interaction strategy.

Norman (2013) emphasizes that effective design should help users understand both the actions that are available and the consequences of those actions. Concepts such as visibility, feedback, mapping, constraints, and conceptual consistency therefore remain central to contemporary interface design.

Mental models have become even more significant with AI-enabled systems. Traditional interfaces generally produce relatively deterministic outcomes: selecting the same control under the same conditions typically produces the same result. Intelligent systems, by contrast, may generate probabilistic or context-dependent responses. Users therefore develop mental models not only about interface controls but also about the capabilities, limitations, uncertainty, and reliability of the underlying AI.

Incorrect mental models of AI can result in both excessive trust and inappropriate rejection. Users who assume that an AI system possesses capabilities beyond its actual competence may rely too heavily on its output. Conversely, users who misunderstand system limitations may reject useful recommendations. Recent HCI research therefore emphasizes trust calibration, in which user reliance should correspond to actual system capability rather than simply maximizing trust. Reviews of AI-enabled systems show that trust is influenced by system characteristics, design features, socio-ethical concerns, and individual user characteristics.

2.3. Principles of Effective Interaction Design

A number of foundational HCI principles remain highly relevant despite substantial technological change. Rather than treating these principles as isolated historical rules, they can be understood as mechanisms for reducing uncertainty and supporting effective human action.

Visibility requires that important system functions and states be perceivable. Users should be able to determine what actions are possible and understand the current status of an ongoing process.

Feedback provides information regarding the consequences of user actions. Immediate and meaningful feedback confirms that an input has been received and allows users to determine whether the resulting state corresponds to their intention.

Consistency enables users to transfer previously acquired knowledge to new parts of a system. Similar operations should behave similarly, while visual and conceptual conventions should remain stable unless a meaningful reason exists for variation.

Mapping concerns the relationship between controls and their effects. Interfaces with natural or easily understood mappings reduce the cognitive effort required to predict system behavior.

Constraints limit inappropriate actions and thereby prevent errors before they occur. Constraints may be physical, logical, semantic, or interface-based.

User control ensures that users retain meaningful authority over interaction. Undo functions, cancellation mechanisms, adjustable settings, confirmation mechanisms, and transparent automation can all contribute to perceived and actual control.

Error prevention and recovery recognize that errors are inevitable components of human interaction. Interfaces should therefore minimize predictable mistakes and provide understandable mechanisms for recovery.

These principles can be observed across several generations of HCI, although their implementation changes depending on the dominant interaction paradigm.

Table 1. Comparison of classical and emerging human–computer interaction paradigms.

Interaction Paradigm	Typical Input	System Role	User Role	Primary HCI Concern
Command-based interaction	Keyboard commands	Computational tool	Operator	Memorability and accuracy
Graphical interaction	Mouse and keyboard	Interactive application	Direct manipulator	Learnability and consistency
Web-based interaction	Navigation and forms	Information environment	Information seeker	Information architecture
Mobile and ubiquitous interaction	Touch, gesture, sensors	Context-aware service	Mobile user	Context and attention
Immersive interaction	Gesture, gaze, voice	Spatial environment	Participant	Presence and cognitive comfort
Human–AI interaction	Natural language and multimodal input	Intelligent collaborator	Supervisor/collaborator	Trust, transparency, and control

Table 1 illustrates an important transition in HCI. As systems have evolved, the role of the computer has gradually changed from a passive execution mechanism toward an active and increasingly adaptive participant in interaction. At the same time, the user has moved from explicitly controlling individual operations toward supervising, interpreting, and collaborating with computational systems. This transformation does not eliminate traditional usability principles; instead, it extends them into new dimensions such as transparency, automation control, and appropriate trust.

2.4. Cognitive Load and Interaction Complexity

Human cognitive resources are limited, and interactive systems compete for these resources during task performance. Cognitive load broadly refers to the mental effort associated with processing information and performing a task. When interface complexity exceeds the user's available cognitive capacity, performance may decrease and errors may increase.

Cognitive workload has therefore become an increasingly important variable in contemporary HCI research. Kosch et al. (2023) emphasize that the growing number of computing systems competing for human attention makes cognitive workload a critical component of interactive-system evaluation. Their survey also shows that workload can be assessed using subjective, behavioral, performance-based, and physiological measures.

Several interface characteristics can unnecessarily increase cognitive load. These include excessive menu depth, inconsistent terminology, information redundancy, frequent interruptions, poor visual hierarchy, unnecessarily complex navigation, and requirements to remember information across multiple screens. Intelligent interfaces can introduce additional burdens when users must formulate appropriate prompts, interpret uncertain recommendations, compare alternative outputs, or determine whether generated information should be trusted.

Reducing cognitive load does not necessarily mean eliminating complexity from a system. Complex professional tasks may inherently require substantial reasoning. The design objective is instead to minimize unnecessary interaction complexity, thereby preserving users' cognitive resources for the actual task.

Adaptive interfaces offer one potential approach. Such systems can modify information presentation, interaction mechanisms, or levels of assistance based on context or user state. Recent human-factors research on adaptive autonomy emphasizes that adaptation should not be considered solely from the technical perspective; effective adaptive systems must also respond appropriately to human characteristics and workload.

2.5. From Static Interfaces to Adaptive Interaction

Traditional interfaces generally provide approximately the same structure regardless of individual differences or changing user conditions.

Contemporary systems increasingly challenge this assumption through adaptive and personalized interaction.

An adaptive interface may modify content organization, recommendation priority, information density, navigation structure, notification frequency, or assistance according to user characteristics and context. Adaptation can be based on explicit preferences, previous interaction history, environmental conditions, behavioral patterns, or inferred cognitive state.

This creates an important HCI trade-off. Adaptation can reduce effort by anticipating user needs, but excessive or unpredictable adaptation may disrupt learned behavior and weaken user control. A system that continuously reorganizes itself may technically optimize certain metrics while simultaneously increasing uncertainty.

For this reason, successful adaptive interaction requires a balance between automation and predictability. Users should be able to understand why important changes occur and retain sufficient control to modify or reject system behavior. This issue becomes particularly important in intelligent systems, where adaptation may be generated dynamically by machine-learning models rather than predefined interface rules.

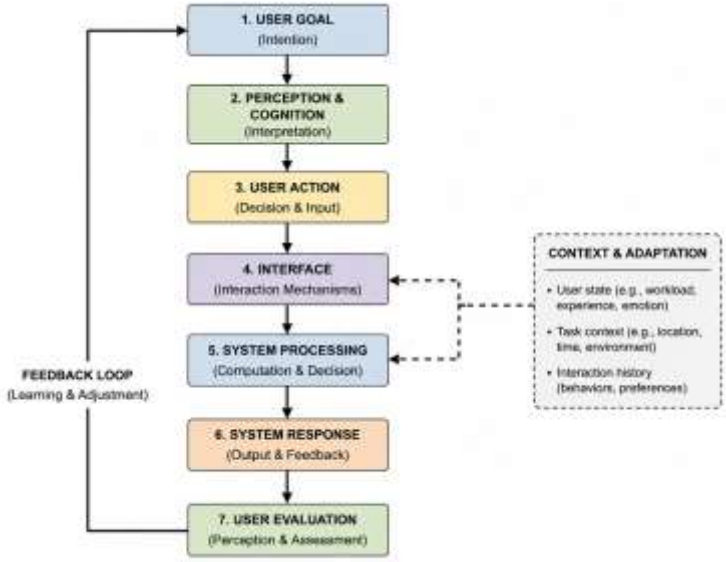


Figure 2. Human-centered interaction framework connecting user goals, cognitive processes, interface mechanisms, system responses, and adaptive feedback.

Figure 2 conceptualizes interaction as a continuous feedback process. A user begins with a goal that is interpreted through perceptual and cognitive processes. The interface translates user actions into system input, after which the computational system generates a response. The user subsequently perceives and evaluates this response, producing feedback that influences the next interaction cycle. In adaptive or intelligent systems, contextual information and previous behavior may additionally modify the system's subsequent response.

This framework highlights why modern HCI cannot be reduced to interface appearance alone. Interaction quality emerges from the complete relationship among the user, interface, system behavior, task context, and feedback mechanisms.

2.6. Human Performance and Interaction Evaluation

The effectiveness of an interactive system can be evaluated through both objective and subjective measures. Objective indicators commonly include task completion time, task success rate, error frequency, interaction steps, response latency, and behavioral patterns. Subjective measures may include perceived usability, satisfaction, workload, trust, comfort, and perceived control.

No single measure is sufficient to characterize interaction quality. A user may complete a task quickly while experiencing high cognitive workload, or may report high satisfaction despite frequent errors. Consequently, HCI evaluations increasingly combine multiple forms of evidence.

This multidimensional approach is particularly important for intelligent systems. Traditional usability metrics can determine whether users can operate an interface efficiently, but they may not reveal whether users appropriately understand or rely on AI-generated recommendations. Human-centered evaluation must therefore examine not only whether interaction succeeds, but also how users arrive at decisions, how much effort is required, and whether reliance on the system is appropriate.

These foundations provide the conceptual basis for human-centered design and usability engineering. The following section therefore examines how human requirements can be systematically incorporated into iterative design and evaluation processes.

3. HUMAN-CENTERED DESIGN AND USABILITY

Human-centered design (HCD) places users, their goals, capabilities, limitations, and usage contexts at the center of the development process. Rather than designing a technological system first and evaluating its suitability afterward, HCD integrates user requirements throughout analysis, design, implementation, and evaluation. This approach is particularly important for contemporary interactive systems because users may differ substantially in experience, physical ability, cognitive characteristics, technological literacy, and environmental context.

ISO 9241-210:2019 defines human-centered design as an approach to interactive-system development that focuses on users and their needs throughout the system life cycle. The standard emphasizes understanding the context of use, specifying user requirements, developing design solutions, and evaluating those solutions against identified requirements. ISO confirmed this edition again in 2025, and it remains the current version of the standard.

In practice, human-centered design should not be understood as a strictly linear procedure. User requirements may change as prototypes are evaluated, previously unidentified usability problems may become visible, and technological limitations may require alternative design decisions. For this reason, contemporary HCI generally adopts an iterative process in which analysis, design, prototyping, and evaluation are repeatedly connected.

3.1. The Iterative Human-Centered Design Process

The first stage of human-centered design involves understanding the context of use. Designers need to identify who the intended users are, what tasks they perform, which goals they attempt to achieve, and under what environmental or organizational conditions interaction takes place. The same interface may perform differently when used by an expert in a controlled workplace, a novice user on a mobile device, or an individual operating under time pressure.

User research may include interviews, observation, questionnaires, contextual inquiry, task analysis, usage logs, and stakeholder discussions. The objective is not simply to collect users' stated preferences but to understand actual behaviors, difficulties, expectations, and constraints.

The second stage involves defining user and system requirements. These requirements translate observations into measurable design objectives. For

example, instead of specifying that an application should be “easy to use,” designers may establish that users should complete a core task within a particular number of steps, with a defined success rate and acceptable error level.

Design solutions are subsequently developed through sketches, wireframes, interactive prototypes, or functional implementations. Early prototypes allow design alternatives to be explored before substantial development resources are committed. As design maturity increases, prototypes may incorporate realistic interaction behavior and visual structure.

Evaluation then determines whether the proposed design satisfies user requirements. The findings are used to identify weaknesses, revise the interface, and initiate another development cycle.

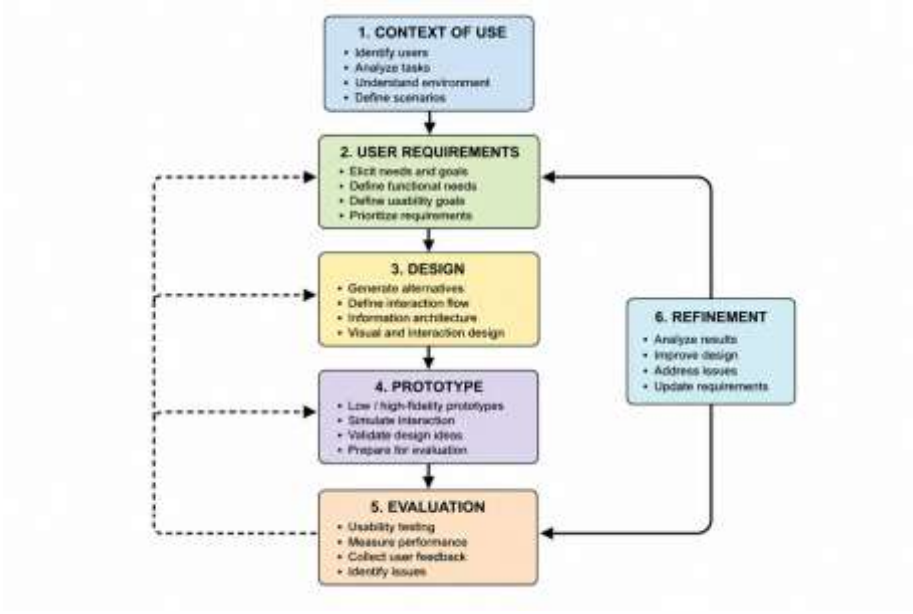


Figure 3. Iterative human-centered design lifecycle integrating context analysis, user requirements, design, prototyping, evaluation, and continuous refinement.

Figure 3 illustrates that human-centered design is not completed after a single evaluation stage. Instead, evaluation results continuously return to earlier phases of development. A usability problem identified during prototype testing may require a design modification, reconsideration of a requirement, or even renewed investigation of the context of use. This feedback structure

reduces the risk of discovering fundamental interaction problems only after the system has been fully implemented.

3.2. Usability as a Multidimensional Quality Attribute

Usability remains one of the central concepts of HCI. Usability has traditionally been evaluated through dimensions such as effectiveness, efficiency, satisfaction, learnability, and error-related measures. Contemporary evaluation retains these dimensions but increasingly considers usability together with accessibility, cognitive workload, trust, and contextual appropriateness.

Effectiveness concerns whether users can achieve intended objectives accurately and completely.

Efficiency concerns the resources required to achieve those objectives, including time, physical effort, cognitive effort, and number of interaction steps.

Satisfaction reflects users' subjective perceptions of comfort, acceptability, confidence, and overall experience.

Learnability refers to how easily new or infrequent users can understand the system and develop effective interaction strategies.

Error tolerance concerns both the probability of user error and the ability of the interface to support recovery when errors occur.

Modern intelligent systems introduce additional dimensions. A system may be easy to operate yet difficult to understand. Similarly, an AI assistant may produce useful recommendations while encouraging excessive reliance. Contemporary usability evaluation must therefore consider whether users understand system capabilities, recognize uncertainty, maintain appropriate control, and can identify incorrect outputs.

Table 2. Key dimensions and representative measures for contemporary usability evaluation.

Evaluation Dimension	Main Question	Representative Measures
Effectiveness	Can users successfully complete the task?	Task success rate, completion accuracy
Efficiency	How much effort is required?	Completion time, interaction steps
Learnability	How easily can the system be learned?	First-attempt success, learning time
Error tolerance	Can users avoid and recover from errors?	Error rate, recovery success
Satisfaction	How do users perceive the experience?	Questionnaires, preference ratings
Cognitive workload	How mentally demanding is interaction?	Workload scales, behavioral measures
Accessibility	Can diverse users operate the system?	Accessibility compliance, task success
Trust and control	Is reliance on system behavior appropriate?	Trust ratings, override behavior

Table 2 demonstrates that usability evaluation should combine observable performance with subjective experience. This becomes particularly important when comparing conventional interfaces with intelligent or adaptive systems because identical task-completion performance may conceal substantial differences in user workload, understanding, or trust.

3.3. Usability Evaluation Methods

Usability evaluation methods can broadly be divided into user-based, expert-based, and data-driven approaches. Combining multiple approaches generally produces a more comprehensive understanding of interaction quality.

User-based testing involves observing representative users while they perform predefined tasks. Measures such as completion time, success rate, errors, navigation behavior, and user comments can be collected. This approach provides direct evidence regarding how actual users interact with the system.

The think-aloud method asks users to verbalize their thoughts while performing tasks. Although verbalization may influence natural behavior to some extent, it can reveal expectations, confusion, decision strategies, and mismatches between user mental models and interface structure.

Heuristic evaluation relies on usability specialists examining an interface according to established interaction principles. It can identify problems relatively early and inexpensively, although expert judgments should not completely replace evaluation involving representative users.

Field studies evaluate technologies within real-world environments. They are particularly useful for mobile, wearable, healthcare, educational, or industrial applications in which contextual conditions strongly influence interaction.

A/B testing compares alternative interface designs using behavioral or performance data. This method is widely used in large-scale digital services, where even relatively small interface modifications may affect user behavior.

Contemporary systems additionally provide large volumes of interaction-log data. Click sequences, dwell time, navigation routes, abandonment points, repeated actions, and error patterns can reveal usability problems that may not appear during controlled laboratory testing. Nevertheless, behavioral data should be interpreted cautiously because it indicates what users do but may not fully explain why they behave in that manner.

3.4. Accessibility and Inclusive Interaction

Accessibility should be integrated into human-centered design from the beginning rather than treated as an additional requirement after development.

The aim is to ensure that interactive technologies can be effectively used by individuals with diverse sensory, motor, cognitive, and situational needs.

Current web accessibility practice is strongly influenced by the Web Content Accessibility Guidelines (WCAG) 2.2. WCAG 2.2 extends earlier versions with additional criteria addressing areas such as focus visibility, dragging movements, target size, consistent help, redundant entry, and accessible authentication. W3C recommends using the most current WCAG version when developing or updating accessibility policies.

These requirements have direct implications for interaction design. For example, controls that require precise dragging may create barriers for users with motor limitations, while small interactive targets can reduce usability on touch-based devices. Similarly, authentication mechanisms that impose unnecessary memory or cognitive demands may exclude users who otherwise have no difficulty accessing the application.

Inclusive design also extends beyond permanent disability. Situational conditions can temporarily alter users' interaction capabilities. Bright outdoor environments may reduce screen visibility, noisy surroundings may prevent effective voice interaction, and carrying an object may restrict interaction to a single hand. Designing for diverse conditions can therefore improve usability for the broader user population.

Accessibility should consequently be considered a component of general interaction quality rather than a separate specialist concern.

3.5. Evaluating Intelligent and Generative Interfaces

AI-based systems require extensions to conventional usability evaluation. Traditional interfaces largely respond to explicit user commands, whereas generative or predictive systems may produce variable results that users must interpret and evaluate.

For generative interfaces, evaluation may therefore examine several additional questions:

- Can users formulate requests that produce appropriate outputs?
- Can users determine when generated information may be incorrect?
- Does the interface communicate system uncertainty or limitations?

- Can users revise, reject, or override system output?
- Does interaction encourage appropriate rather than excessive reliance?
- Does the system support efficient correction when its output is unsuitable?

These questions demonstrate that usability in AI-enabled interaction is partly concerned with the quality of the collaboration between human and system rather than interface efficiency alone.

A generative AI interface may technically respond within seconds, yet users may spend substantial additional time checking its results. Similarly, an interface may appear highly intuitive while failing to communicate uncertainty. Evaluation must therefore capture the complete interaction process, including verification, correction, interpretation, and decision-making.

3.6. Mixed-Method Evaluation

Because interaction has objective, subjective, behavioral, and contextual dimensions, contemporary HCI increasingly benefits from **mixed-method evaluation**.

Quantitative measurements provide comparable evidence regarding performance. Examples include:

- task completion rate,
- completion time,
- number of errors,
- number of interaction steps,
- response latency,
- navigation efficiency.

Qualitative methods provide complementary explanations. Interviews, observations, open-ended questionnaires, and think-aloud protocols can reveal frustration, misunderstanding, unexpected strategies, and contextual factors.

Subjective questionnaires can additionally measure constructs such as perceived usability, workload, satisfaction, trust, and presence.

Combining these forms of evidence reduces the risk of drawing conclusions from a single performance indicator. For example, two interface designs may produce similar completion times while one imposes substantially greater mental workload. Conversely, users may report high satisfaction with a visually appealing interface despite repeatedly making errors.

Human-centered evaluation should therefore answer three complementary questions:

What happened during interaction? Performance and behavioral measurements address this question.

Why did it happen? Qualitative methods help identify underlying causes.

How did the user experience it? Subjective measures describe perception, workload, satisfaction, and trust.

Together, these dimensions provide a more comprehensive basis for interaction design decisions.

3.7. From Usability to User Experience

Usability is essential but does not fully represent the quality of modern interactive systems. The broader concept of user experience (UX) includes emotional responses, expectations, perceived value, aesthetics, engagement, trust, and the meaning associated with technology use.

This distinction is important because a technically usable system may still provide an undesirable experience. Conversely, an engaging or visually sophisticated product cannot compensate indefinitely for poor effectiveness or persistent interaction errors.

Contemporary HCI therefore increasingly treats usability and UX as complementary dimensions. Usability emphasizes whether users can successfully interact with a system, whereas UX considers the broader experience that emerges before, during, and after interaction.

The importance of this distinction increases as computational systems become embedded in everyday activities and social contexts. Intelligent assistants,

immersive systems, wearable devices, and adaptive interfaces may accompany users over extended periods rather than during isolated tasks. Their success therefore depends not only on momentary task performance but also on long-term acceptance, confidence, comfort, and perceived usefulness.

The next section extends this human-centered perspective toward emerging interaction technologies, focusing particularly on artificial intelligence, generative systems, multimodal interaction, and immersive environments.

4. HUMAN INTERACTION WITH INTELLIGENT AND EMERGING SYSTEMS

The rapid development of artificial intelligence, immersive technologies, advanced sensing systems, and multimodal interfaces is transforming the traditional boundaries of Human–Computer Interaction. Earlier interactive systems were primarily designed to receive explicit commands and return predictable responses. Contemporary systems, however, increasingly infer user intentions, generate content, recommend actions, adapt to context, and participate in decision-making processes.

This transformation introduces a fundamental shift in interaction design. The computational system is no longer simply an instrument controlled through an interface; in many applications, it becomes an active participant capable of generating alternatives, interpreting ambiguous input, and adapting its behavior. Consequently, interaction quality increasingly depends on the relationship established between human judgment and computational capability.

4.1. Human–AI Interaction

Human–AI Interaction (HAII) examines how people communicate, cooperate, supervise, understand, and make decisions with artificial intelligence systems. Unlike traditional HCI, where system behavior is largely predefined, AI-enabled systems may produce probabilistic and context-dependent outcomes.

This creates several new design questions. Users need to understand what an AI system can and cannot do, determine when its output should be accepted or verified, and maintain sufficient control over important decisions. Raees et al. (2024) argue that contemporary Human-Centered AI should move beyond explanation alone toward forms of interaction in which users can actively influence, contest, adapt, and collaborate with intelligent systems.

Human–AI interaction can therefore be conceptualized through several levels of involvement. At the lowest level, AI provides information or recommendations while the user remains the primary decision maker. At intermediate levels, humans and AI systems collaboratively evaluate alternatives. At higher levels of automation, the AI system may execute actions autonomously while users supervise or intervene when necessary.

The appropriate balance depends strongly on context. A music recommendation system can tolerate comparatively high levels of automation because an unsuitable recommendation has limited consequences. By contrast, healthcare, transportation, financial decision support, and industrial control systems require stronger mechanisms for transparency, oversight, and human intervention.

AI-assisted interpretation is also becoming increasingly relevant in specialized sensing and engineering domains. For example, Arvas and Çınar (2026) demonstrated the use of a lightweight YOLO-based approach for detecting underground objects from Ground Penetrating Radar (GPR) data. From an HCI perspective, such AI-based detection systems can be considered decision-support components that complement expert interpretation by rapidly processing complex sensor-derived information.

An important design objective is therefore not to maximize automation but to establish appropriate human–AI collaboration (Amershi et al., 2019).

4.2. Generative AI as a New Interaction Paradigm

Generative artificial intelligence has introduced one of the most significant recent transitions in HCI. Large language models and multimodal foundation models allow users to interact through natural-language instructions rather than predefined menus, commands, or rigid interaction sequences.

In conventional interfaces, designers typically determine the available actions in advance. In generative interfaces, users can formulate open-ended requests and potentially obtain a wide range of outputs. This flexibility substantially expands the interaction space but simultaneously transfers part of the interaction-design burden to the user.

For example, successful use of a generative AI system may require users to:

- formulate an appropriate request,

- provide sufficient contextual information,
- evaluate generated output,
- detect potential errors or unsupported claims,
- refine the request when necessary,
- compare alternative outputs,
- and determine whether external verification is required.

The interaction consequently becomes iterative rather than transactional.

A typical sequence may involve an initial request, generated response, user evaluation, refinement, regeneration, and eventual acceptance or rejection. This resembles a collaborative dialogue more closely than conventional command execution.

Generative AI systems also introduce the problem of output variability. Identical or similar requests may produce different responses, and fluent presentation does not necessarily indicate factual reliability. Interface design should therefore support revision, traceability, comparison, verification, and user correction.

Human–AI collaboration research also indicates that generative systems may reduce cognitive burden in information-intensive tasks while simultaneously creating risks of over-reliance. Consequently, transparency, accountability, and meaningful human oversight remain important components of collaborative system design.

4.3. Trust, Transparency, and Appropriate Reliance

Trust is one of the most important factors influencing interaction with intelligent systems. However, the objective of HCI should not be to maximize user trust indiscriminately. Instead, the goal is calibrated trust, in which the level of reliance approximately corresponds to the actual capabilities and limitations of the system.

If trust is too low, users may ignore beneficial automation. If trust is too high, they may accept inappropriate or incorrect system outputs without sufficient verification.

Several interface characteristics influence trust formation:

- consistency of system behavior,
- clarity of feedback,
- visibility of uncertainty,
- explanation of important recommendations,
- ability to correct or override system actions,
- disclosure of limitations,
- and previous interaction experience.

Transparency does not necessarily require exposing every technical detail of an algorithm. Excessive information may itself increase cognitive workload. Effective transparency instead requires providing users with information that is relevant to the current task and sufficient to support informed decisions.

For example, an intelligent decision-support interface may indicate the basis of a recommendation, communicate confidence or uncertainty where appropriate, allow alternative possibilities to be inspected, and provide a mechanism for human override.

These mechanisms contribute to an interaction process in which users remain active participants rather than passive recipients of algorithmic output.

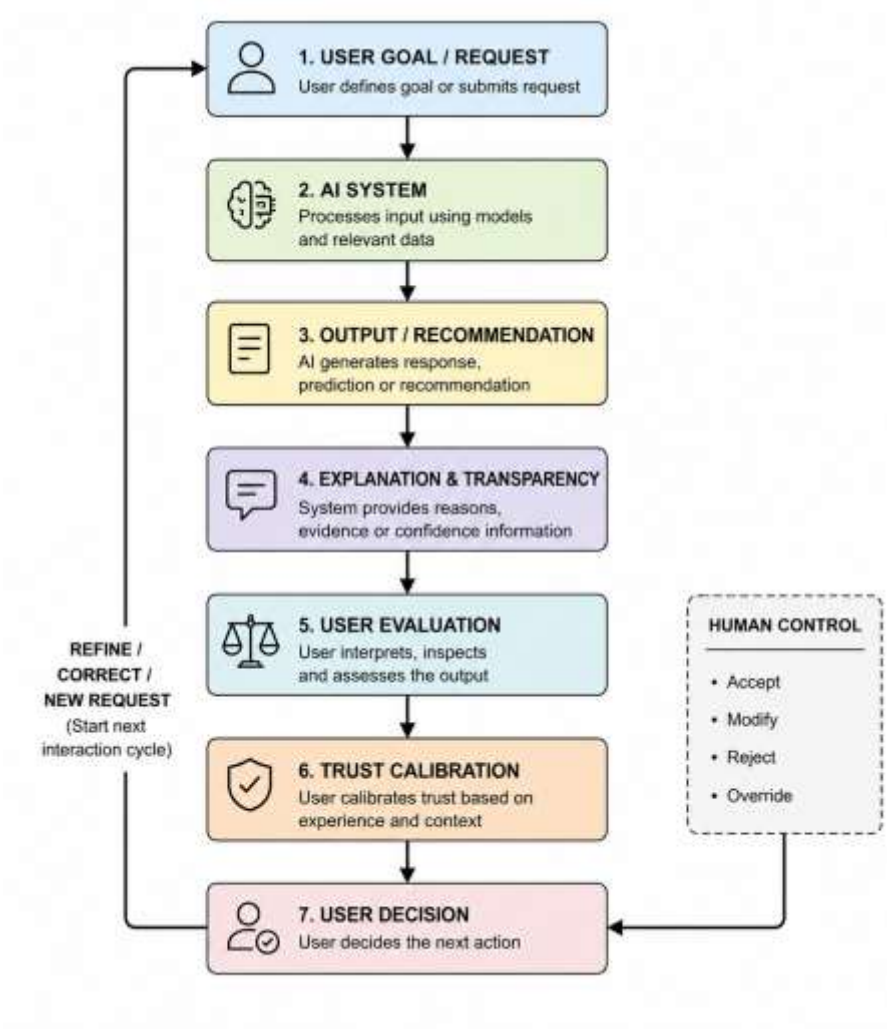


Figure 4. Human–AI interaction loop integrating user goals, AI processing, generated outputs, explanation, evaluation, trust calibration, and iterative feedback.

Figure 4 illustrates Human–AI interaction as a continuous collaborative cycle. The process begins with a user goal or request and continues through AI processing and generation of an output, prediction, or recommendation. Rather than immediately terminating after system output, the interaction proceeds through explanation, user evaluation, and trust calibration. The user may then accept, modify, reject, or refine the system response, thereby initiating another interaction cycle.

This structure distinguishes intelligent interaction from conventional one-directional automation. Human feedback can influence subsequent system behavior, while system output modifies the user's understanding and decisions.

4.4. Multimodal Human–Computer Interaction

Human communication is naturally multimodal. People combine speech, visual information, facial expression, gesture, gaze, body movement, touch, and contextual cues during everyday interaction. Traditional computer interfaces, however, have historically relied on comparatively limited input channels such as keyboards, pointing devices, and touchscreens.

Multimodal HCI seeks to reduce this mismatch by enabling computational systems to receive and integrate information through several interaction channels.

Typical modalities include:

Speech: Natural-language commands and conversational interaction.

Gesture: Hand or body movements captured through cameras or motion sensors.

Gaze: Eye-tracking information used for attention estimation, selection, or navigation.

Touch: Direct manipulation through touchscreens and tactile interfaces.

Facial expression: Visual signals that may support affect-aware interaction.

Physiological signals: Measurements such as heart rate or related signals that may contribute to estimating workload or user state.

Visual input: Images, video, objects, and environmental information.

Recent multimodal systems increasingly employ deep-learning and transformer-based architectures to combine information from multiple channels. A recent systematic review of multimodal HCI identifies a transition from rule-based and conventional fusion mechanisms toward deep learning and transformer-based approaches capable of modeling richer relationships among modalities (Thushara et al., 2026).

The principal advantage of multimodal interaction is not simply the availability of more input methods. Different modalities can complement one another. Speech may be convenient when hands are occupied, gaze may indicate an object of interest, and gesture may specify a spatial relationship that would be difficult to communicate verbally.

Multimodal systems can therefore support more flexible and context-sensitive interaction. Nevertheless, they also introduce challenges such as ambiguous signals, synchronization between modalities, sensor reliability, privacy, computational complexity, and user variability.

4.5. Immersive and Spatial Interaction

Virtual Reality (VR), Augmented Reality (AR), and Mixed Reality (MR) extend interaction beyond two-dimensional screens into three-dimensional environments.

In Virtual Reality, users interact within a predominantly computer-generated environment.

In Augmented Reality, digital information is superimposed on the physical environment.

In Mixed Reality, digital and physical elements can become more deeply integrated and interact with one another within a shared spatial context.

These technologies alter several fundamental HCI assumptions. Instead of clicking a two-dimensional interface element, users may point toward, look at, approach, manipulate, or speak to a virtual object. Consequently, interaction becomes increasingly embodied and spatial.

Immersive visualization research emphasizes the importance of multimodal perception and interaction in these environments. Spatial interfaces may integrate vision, movement, hand interaction, gaze, audio, and haptic feedback to produce a stronger sense of presence and engagement.

However, immersive systems introduce new usability concerns. These include:

- motion sickness and cybersickness,
- physical fatigue,

- limited field of view,
- spatial disorientation,
- inaccurate tracking,
- interaction precision,
- privacy in continuously sensed environments,
- and accessibility for users with different physical capabilities.

Interface elements must also account for depth, viewing distance, physical reach, and three-dimensional positioning. A design principle that works effectively on a desktop display may therefore be inappropriate in an immersive environment.

4.6. Context-Aware and Adaptive Interaction

Modern interactive systems increasingly use contextual information to determine how and when services should be presented. Context may include location, time, device state, environmental conditions, previous interaction history, user activity, and inferred preferences.

A context-aware system can modify its behavior according to these conditions. For example, information density may be reduced when the user is moving, notifications may be delayed when workload is high, or interface components may be reorganized according to frequently performed tasks.

AI further strengthens this capability by allowing systems to infer patterns from previous interactions and generate personalized responses.

Nevertheless, personalization introduces an important tension between convenience and user autonomy. If adaptation occurs invisibly, users may struggle to understand why an interface behaves differently over time. Excessive personalization may also reduce predictability or expose users to unwanted inference about their behavior.

Adaptive interaction should therefore satisfy three principles:

1. **Predictability:** Users should maintain a reasonable understanding of system behavior.

2. **Controllability:** Users should be able to modify or disable important adaptations.

3. **Transparency:** Meaningful adaptation should be understandable when it influences important decisions.

These principles become increasingly important as interfaces evolve toward proactive rather than purely reactive behavior.

4.7. Comparison of Emerging Interaction Technologies

The emerging technologies discussed in this section differ substantially in their interaction mechanisms, opportunities, and design challenges.

Table 3. Emerging HCI technologies, interaction mechanisms, opportunities, and principal design challenges.

Technology	Primary Interaction Modes	Main Opportunity	Major HCI Challenge
Generative AI	Natural language, image, speech	Flexible human–AI collaboration	Reliability and over-reliance
Conversational AI	Speech and text	Natural communication	Context and intent understanding
Multimodal systems	Voice, gaze, gesture, touch	Flexible and accessible interaction	Modality fusion
Augmented Reality	Gaze, gesture, spatial input	Physical–digital integration	Registration and usability
Virtual Reality	Motion, gesture, gaze	Immersion and presence	Cybersickness and fatigue
Wearable interaction	Sensors, touch, physiological input	Continuous contextual interaction	Privacy and comfort

Technology	Primary Interaction Modes	Main Opportunity	Major HCI Challenge
Adaptive interfaces	Behavioral/contextual data	Personalization	Predictability and user control

Table 3 demonstrates that no emerging interaction technology provides universal advantages. Each technology introduces new capabilities while simultaneously creating specific human-factors challenges. The selection of an interaction paradigm should therefore depend on task characteristics, user requirements, environmental conditions, and risk level rather than technological novelty alone.

4.8. Toward Collaborative and Symbiotic Interaction

The overall trajectory of HCI suggests a gradual movement from tool-oriented interaction toward collaborative interaction.

In traditional computing, users issued commands and computers executed those commands. In contemporary intelligent environments, computational systems increasingly suggest actions, infer goals, generate alternatives, and participate in problem solving.

Recent Human–AI Interaction research identifies collaboration, competition, conflict, and symbiosis among the emerging forms of relationships between humans and AI systems. However, the most desirable relationship depends on the specific application and cannot be defined purely by increasing system autonomy.

The future of interaction is therefore unlikely to involve the replacement of humans by increasingly autonomous interfaces in every domain. A more plausible direction is the development of systems in which computational capabilities complement human judgment, creativity, contextual understanding, and responsibility.

From this perspective, successful HCI will increasingly depend on designing the relationship between human and machine, rather than designing the interface as an isolated technological component.

The next section therefore examines the ethical and societal conditions required for this relationship, with particular emphasis on trust, privacy, fairness, accessibility, user autonomy, and responsible interaction design.

5. TRUST, ETHICS, PRIVACY, AND INCLUSIVE INTERACTION

As interactive technologies become increasingly intelligent, personalized, and data-driven, the quality of interaction can no longer be evaluated exclusively through conventional usability measures. Contemporary HCI must also consider whether systems deserve user trust, preserve meaningful autonomy, protect personal information, avoid discriminatory outcomes, and remain accessible to diverse populations.

These concerns become particularly significant when computational systems influence decisions rather than merely support task execution. Recommendation systems, conversational agents, automated decision-support tools, adaptive interfaces, and generative AI applications can shape what information users encounter and how they interpret available alternatives. Consequently, interaction design carries ethical responsibilities that extend beyond visual appearance or functional efficiency.

5.1. Trust as a Dynamic Interaction Property

Trust represents a central factor in the adoption and continued use of interactive technologies. However, trust in HCI should not be regarded as a fixed user attitude. It develops over time through experience with system performance, feedback, reliability, transparency, and contextual conditions.

Recent systematic research demonstrates that trust in HCI is multidimensional and emerges from the complex interaction of psychological, technological, and contextual factors (Gulati et al., 2024).

This is especially important for AI-enabled systems. Users may initially approach an unfamiliar intelligent system with uncertainty. Repeated accurate outcomes can gradually increase reliance, while unexpected or incorrect outputs may decrease confidence. Trust can therefore change across interaction episodes.

Two problematic situations can emerge:

Under-trust occurs when users reject or ignore a system even when its performance is sufficiently reliable.

Over-trust occurs when users accept system outputs without appropriate scrutiny, including situations in which the system may be incorrect.

The design objective should therefore be appropriate or calibrated trust, rather than maximum trust.

Interfaces can support calibrated trust by communicating system capabilities and limitations, providing meaningful feedback, revealing uncertainty where relevant, and allowing users to inspect or reconsider important system outputs. Users should also be able to intervene when automation does not correspond to their goals.

5.2. Transparency and Explainability

Transparency concerns the extent to which users can form an appropriate understanding of system behavior. In conventional applications, transparency may involve showing current system status, confirming actions, or explaining the consequences of an interface operation. Intelligent systems introduce additional requirements because their internal decision processes may be complex or probabilistic.

Explainability has therefore become an important component of Human–AI Interaction. However, providing more information does not automatically improve interaction. Highly technical explanations may increase cognitive workload without helping users make better decisions.

The objective should instead be task-relevant transparency. Users should receive the information necessary to answer practical questions such as:

- Why did the system produce this recommendation?
- What information influenced the result?
- How certain is the system?
- What are its known limitations?
- Can the recommendation be changed or rejected?
- What should the user verify before acting?

This approach shifts explainability from algorithmic description toward decision support.

Transparency also requires appropriate timing. Information presented too early may distract users, while explanations presented only after an error may arrive too late. Designers must therefore determine what information should be visible by default and what should remain available through progressive disclosure.

5.3. Privacy-Aware Interaction Design

Modern interactive systems frequently rely on user data for personalization, adaptation, recommendation, and prediction. These capabilities may improve convenience, but they also create significant privacy challenges.

Data may be collected through explicit user input, interaction histories, location information, device sensors, browsing behavior, biometric signals, voice recordings, or inferred preferences. In many systems, users may not fully understand how these data are collected or how they influence subsequent system behavior.

Privacy-aware HCI should therefore make important data practices understandable and controllable.

Users should be able to determine:

1. what information is being collected,
2. why it is required,
3. how it will be used,
4. whether it will be shared,
5. how long it will be retained,
6. and whether consent can later be withdrawn.

Consent mechanisms should also avoid unnecessary complexity. Interfaces that technically provide a choice while making privacy-preserving options difficult to discover do not provide meaningful control.

This issue becomes increasingly important in immersive and multimodal systems. Cameras, microphones, eye trackers, wearable sensors, and spatial-mapping technologies may continuously collect information about both users and their environments. Consequently, privacy design must increasingly address persistent sensing rather than isolated data-entry events.

5.4. Deceptive and Manipulative Interface Design

An ethically problematic interface may remain technically usable while influencing users toward choices that primarily benefit the service provider. Such practices are commonly discussed in HCI research under the concepts of deceptive design or dark patterns.

Examples can include visually emphasizing one option while obscuring another, making subscription cancellation substantially more difficult than registration, preselecting privacy-invasive choices, introducing artificial urgency, or repeatedly prompting users until they accept a preferred action.

Recent HCI research has moved beyond simply cataloguing such patterns. Caragay et al. (2024), for example, propose evaluating design behavior against users' legitimate expectations and argue that problematic design can arise when the underlying behavior of an application differs from what users reasonably expect.

This perspective is important because manipulation may exist not only in visual presentation but also in underlying application behavior. Excessive notifications, persistent engagement prompts, or defaults that systematically favor the service provider may influence users even when individual interface elements appear conventional.

Recent work has therefore emphasized user empowerment alongside detection. Interventions may help users identify manipulative patterns and understand the intentions behind them rather than relying exclusively on regulation or designer awareness.

Ethical interaction design should consequently preserve meaningful choice rather than merely provide nominal choice.

5.5. Fairness and Algorithmic Bias

AI-enabled interfaces may reproduce or amplify biases contained in training data, system design decisions, or deployment contexts. These issues become especially significant when systems influence access to opportunities, recommendations, information, or important services.

From an HCI perspective, fairness is not exclusively an algorithmic property. The way system outputs are presented can also affect user interpretation.

For example, an interface that presents an automated recommendation as definitive may encourage users to overlook uncertainty or contextual limitations. Similarly, ranking mechanisms can make certain alternatives substantially more visible than others, even when multiple options are technically available.

Human-centered fairness therefore requires consideration of both computational outcomes and interface presentation.

Designers should consider:

- whether different user groups experience comparable interaction quality,
- whether system errors disproportionately affect particular populations,
- whether users can contest automated outcomes,
- whether recommendations are presented with appropriate uncertainty,
- and whether design choices unintentionally reinforce existing inequalities.

In high-impact systems, mechanisms for human review and appeal become particularly important.

5.6. User Agency and Meaningful Human Control

User agency refers to the ability of individuals to intentionally influence technological behavior rather than merely respond to decisions already made by the system.

Agency can be weakened when automation is difficult to understand, difficult to disable, or presented as unavoidable. Adaptive interfaces may similarly reduce agency when system changes occur without explanation.

Meaningful human control can be supported through mechanisms such as:

- configurable automation,
- confirmation before high-impact actions,
- undo and recovery,
- alternative recommendations,
- manual override,
- editable AI-generated content,
- and understandable preference controls.

However, providing a large number of controls does not automatically create greater agency. Excessive configuration may itself increase complexity. Effective control should therefore focus on decisions that materially affect user outcomes.

In intelligent systems, an especially important distinction exists between assistance and substitution. Systems that support users in evaluating alternatives may preserve human judgment, whereas systems that silently replace user decisions can weaken autonomy.

5.7. Inclusive and Accessible Intelligent Systems

Inclusive HCI seeks to ensure that technological systems accommodate variation among users rather than designing exclusively for an assumed average individual.

User populations may differ in visual, auditory, motor, cognitive, linguistic, educational, and technological capabilities. These differences can interact with age, context, device characteristics, and environmental conditions.

AI-based systems create both opportunities and risks for accessibility.

Potential opportunities include:

- automatic speech transcription,
- image descriptions,
- language simplification,
- alternative input modalities,
- personalized interface presentation,
- and adaptive assistance.

At the same time, AI-generated accessibility features may introduce errors. Incorrect image descriptions, inaccurate speech recognition, or poorly generated simplified text can create new barriers if users are expected to rely on them without verification.

Accessibility must therefore remain an explicit design objective rather than being assumed to emerge automatically from intelligent functionality.

Inclusive design should also account for situational impairment. A person using a device while carrying an object, interacting under bright sunlight, operating in a noisy environment, or experiencing temporary physical limitations may benefit from features originally introduced for accessibility.

This illustrates an important HCI principle: designs that accommodate human diversity frequently improve interaction for the broader population.

5.8. Responsible Interaction Design Framework

The previous sections demonstrate that responsible HCI requires simultaneous consideration of multiple dimensions. Usability alone cannot determine whether an interactive system serves users appropriately.

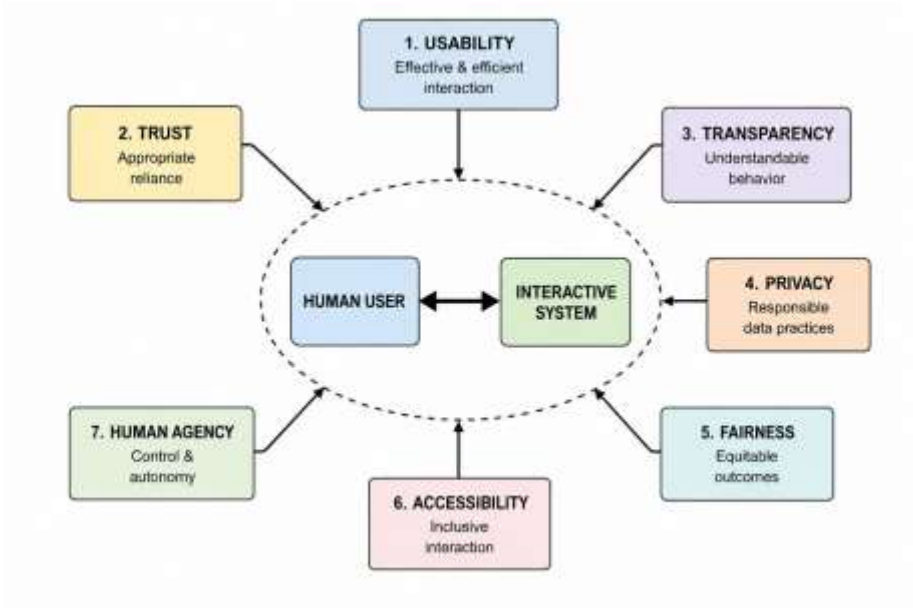


Figure 5. Responsible human–computer interaction framework integrating usability, trust, transparency, privacy, fairness, accessibility, and human agency.

Figure 5 conceptualizes responsible interaction design around a central relationship between the human user and interactive system. Seven surrounding dimensions influence this relationship: usability, trust, transparency, privacy, fairness, accessibility, and human agency.

These dimensions should not be treated independently. Increasing transparency may improve trust but can also increase cognitive load if excessive technical information is presented. Personalization may improve usability while simultaneously increasing privacy risks. Automation may improve efficiency while reducing perceived control. Responsible HCI therefore involves managing trade-offs rather than optimizing a single criterion.

The framework can be summarized through four broader objectives:

Human capability: The system should correspond to perceptual, cognitive, and physical characteristics.

Human understanding: Users should be able to develop an appropriate understanding of system behavior.

Human control: Important decisions should preserve meaningful mechanisms for intervention and correction.

Human values: Design should respect privacy, fairness, accessibility, dignity, and autonomy.

These objectives become increasingly important as interactive systems gain greater autonomy and influence over user decisions.

5.9. From Ethical Guidelines to Design Practice

One persistent challenge in responsible technology development is translating high-level ethical principles into concrete interface decisions. Concepts such as fairness, transparency, autonomy, and privacy are valuable, but designers require mechanisms for determining how these principles affect actual interactions.

Recent research on deceptive design illustrates this transition. Rather than simply telling designers which practices to avoid, emerging frameworks attempt to define positive expected behaviors that can guide concrete software design choices. Caragay et al. (2024) explicitly propose such a concept-based approach to ethical software design.

A practical responsible-design process can therefore ask:

Before interaction:
What information does the user need in order to make an informed choice?

During interaction:
Does the interface preserve meaningful options and clearly communicate system behavior?

After interaction:
Can the user inspect, reverse, correct, or challenge the outcome where appropriate?

Applying these questions throughout development shifts ethics from a final compliance check toward an integral component of human-centered design.

The resulting perspective is particularly important for intelligent and adaptive technologies. As systems increasingly mediate information, recommendations, communication, and decisions, HCI must address not only

how effectively humans interact with technology but also whether those interactions support informed, autonomous, and equitable participation.

The final section builds on these principles to examine emerging directions in Human–Computer Interaction and the future relationship between humans and increasingly intelligent computational environments.

6. FUTURE DIRECTIONS AND CONCLUSION

Human–Computer Interaction continues to evolve as computational technologies become more intelligent, context-aware, immersive, and closely integrated with everyday human activity. The next stage of HCI is unlikely to be defined by a single interface technology. Instead, future interaction environments will increasingly combine artificial intelligence, multimodal sensing, spatial computing, wearable technologies, adaptive interfaces, and potentially direct neural interaction.

This evolution also changes the fundamental research question of HCI. Earlier generations of interface research frequently concentrated on how users could operate computers more efficiently. Future HCI must increasingly investigate how humans and computational systems can cooperate, adapt, negotiate control, and make decisions together while preserving human autonomy, accessibility, safety, and social responsibility.

Recent research confirms this transition. A 2025 CHI review of future-oriented HCI literature argues that future HCI should move beyond narrowly technology-driven predictions and place greater emphasis on uncertainty, human experience, social consequences, and alternative technological futures (Sanchez et al., 2025).

6.1. Human–AI Collaboration and Hybrid Intelligence

Human–AI collaboration is expected to become one of the dominant themes of future interactive-system research. Instead of conceptualizing AI exclusively as an autonomous replacement for human activity, HCI increasingly investigates arrangements in which human and computational capabilities complement one another.

Humans contribute contextual understanding, social knowledge, ethical judgment, creativity, and responsibility. AI systems can contribute computational scale, pattern recognition, rapid information processing, and

generation of alternative solutions. Effective collaboration therefore depends on allocating tasks according to these complementary strengths.

This perspective has encouraged interest in hybrid intelligence, where human and artificial intelligence operate as components of a combined problem-solving system.

However, effective collaboration requires more than connecting users to powerful AI models. Systems need appropriate mechanisms for communication, feedback, intervention, correction, and negotiation of responsibility. The 2026 systematic review by Seini et al. identifies changing human–machine relationships, effective interaction design, and broader organizational and societal implications as major research themes for future Human–AI collaboration.

Raees et al. (2024) similarly argue that future Human-Centered AI should provide users with greater agency than conventional explainable-AI approaches, allowing people not only to understand AI outputs but also to interact with and influence intelligent-system behavior.

Accordingly, future HCI will increasingly need to address a fundamental question:

Which decisions should be automated, which should remain under direct human control, and which should emerge through genuine human–AI collaboration?

There is unlikely to be a universal answer. Appropriate allocation will depend on uncertainty, expertise, risk, task complexity, and the consequences associated with incorrect decisions.

6.2. Adaptive and Proactive Interfaces

Traditional interfaces usually wait for explicit user actions. Future systems will increasingly operate proactively by predicting needs, recognizing context, and adapting before users directly request assistance.

An adaptive system may modify:

- content presentation,
- information density,

- interaction modality,
- notification timing,
- recommended actions,
- navigation structure,
- or level of automated assistance.

Future adaptation may also respond to inferred user states such as cognitive workload, fatigue, attention, or task difficulty.

This capability offers significant opportunities. For example, an interface may reduce unnecessary information during cognitively demanding tasks or provide additional guidance when user difficulty is detected.

However, adaptation creates an important design tension between personalization and predictability. Interfaces that change too frequently may become difficult to learn, while invisible personalization may prevent users from understanding why particular information is presented.

Future adaptive systems should therefore incorporate mechanisms for transparency and controllability. Users should be able to recognize meaningful adaptations and retain authority over significant changes.

The objective should not simply be to create interfaces that adapt to humans, but to establish mutual adaptation, in which both user and system progressively develop effective interaction strategies.

6.3. Multimodal and Natural Interaction

Future interaction will increasingly move beyond reliance on a single input modality. Voice, gaze, gesture, touch, facial expression, movement, environmental sensing, and natural language can potentially be integrated into unified interaction environments.

The principal benefit of multimodal interaction is flexibility. Users may select or combine interaction channels according to task and context.

For example:

- voice may be preferable when hands are occupied,

- gaze may identify an object of interest,
- gesture may communicate spatial relationships,
- touch may provide precise direct manipulation,
- and natural language may communicate complex intent.

Generative AI and multimodal foundation models further strengthen this direction by enabling systems to jointly interpret text, images, speech, and other forms of information.

Consequently, future interfaces may increasingly resemble natural human communication rather than conventional graphical interface operation.

However, multimodal interaction also increases system complexity. Signals may contradict one another, sensors may fail, user behavior can vary substantially, and continuous sensing may create privacy concerns.

Future HCI research must therefore examine how multiple modalities can be combined without creating unnecessary cognitive or technological complexity.

6.4. Spatial Computing and Extended Reality

Spatial computing represents another important direction in which digital interaction becomes integrated with three-dimensional physical environments.

Virtual Reality, Augmented Reality, and Mixed Reality already allow users to manipulate digital information through spatial movement, gaze, hand tracking, and embodied interaction. Future systems are likely to strengthen this integration through more accurate environmental sensing, wearable displays, and persistent digital objects.

Instead of interacting primarily through a rectangular display, users may increasingly operate within environments in which digital information appears directly around physical objects and locations.

Potential application areas include:

- medical visualization,
- education and training,

- industrial maintenance,
- architectural design,
- remote collaboration,
- scientific visualization,
- and entertainment.

The HCI challenges of spatial computing differ considerably from those of conventional interfaces. Designers must consider physical movement, reachability, depth perception, visibility, spatial memory, fatigue, and environmental safety.

Long-term use also raises questions regarding social acceptance, privacy, and the psychological consequences of continuously combining physical and digital environments.

Future spatial interfaces must therefore be evaluated not only according to technological immersion but also according to comfort, accessibility, social appropriateness, and sustainable use.

6.5. Affective and Emotion-Aware Interaction

Another emerging direction is the development of systems capable of recognizing or responding to aspects of human emotional state.

Affective computing may use facial expressions, speech characteristics, physiological signals, interaction behavior, or combinations of multiple signals to infer emotional or cognitive conditions.

Potential applications include adaptive education, healthcare assistance, driver monitoring, entertainment, and personalized digital services.

However, emotion-aware interaction introduces major methodological and ethical challenges. Human emotions are contextual, culturally influenced, and difficult to infer reliably from isolated signals. Incorrect emotional inference may lead systems to make inappropriate adaptations.

Furthermore, emotion data can be highly sensitive.

Future affective systems should therefore avoid treating algorithmic emotion estimates as objective representations of internal human states. They should instead communicate uncertainty and preserve user control over how affective information is collected and used.

This area also intersects with Brain–Computer Interfaces. A recent review of EEG-based affective BCIs identifies considerable progress in emotion recognition and regulation while emphasizing persistent challenges associated with inter-subject variability and real-world deployment (Chen et al., 2025).

6.6. Brain–Computer Interaction

Brain–Computer Interfaces (BCIs) represent one of the most technically ambitious future directions in HCI.

BCI systems seek to establish communication between neural activity and computational systems, potentially allowing users to interact without conventional physical input mechanisms.

Research has explored applications involving:

- assistive communication,
- motor rehabilitation,
- neuroprosthetics,
- control of external devices,
- cognitive-state estimation,
- and affective interaction.

For individuals with severe motor impairments, such technologies may provide forms of digital interaction that would otherwise be inaccessible.

However, widespread BCI use continues to face substantial technical and human-factors challenges. Neural signals can be noisy and highly variable between users and sessions. Interaction accuracy, training requirements, device comfort, long-term reliability, privacy, and user acceptance remain important issues.

Neural data also introduce particularly sensitive privacy questions. Unlike conventional interaction logs, neural signals may potentially reveal information related to cognitive or affective states. Future HCI research must therefore address neuroprivacy, informed consent, data ownership, and user autonomy alongside technical performance.

BCIs should consequently be understood not simply as novel input devices but as technologies that may redefine the boundary between human cognition and computational systems.

6.7. Digital Well-Being and Sustainable Interaction

As technologies become increasingly persistent in everyday life, future HCI will need to consider not only whether users can interact with systems effectively but also whether continued interaction supports long-term well-being.

Interfaces designed exclusively to maximize engagement may encourage excessive use, unnecessary interruption, or attentional competition. Personalized AI systems may intensify these effects by continuously optimizing content according to individual behavioral patterns.

Human-centered design should therefore increasingly incorporate digital well-being as a design objective.

A 2025 systematic review on Human-Centered AI and digital well-being emphasizes the need to integrate AI design with user autonomy, transparency, ethical responsibility, and long-term well-being rather than evaluating systems only through performance or engagement indicators (Shin, 2025).

Future interaction design may therefore incorporate mechanisms such as:

- attention-sensitive notifications,
- controllable personalization,
- interaction-time awareness,
- transparent recommendation mechanisms,
- user-defined limits,
- and reduced unnecessary engagement.

Sustainability also extends to environmental considerations. Increasing computational complexity, large AI models, immersive hardware, and continuously connected devices consume substantial resources. Sustainable HCI can therefore include both human well-being and the environmental consequences of technological interaction.

6.8. Future Human–Computer Interaction Ecosystem

The technological directions discussed above should not be considered independent developments. Future interaction environments are likely to integrate several of these technologies simultaneously.

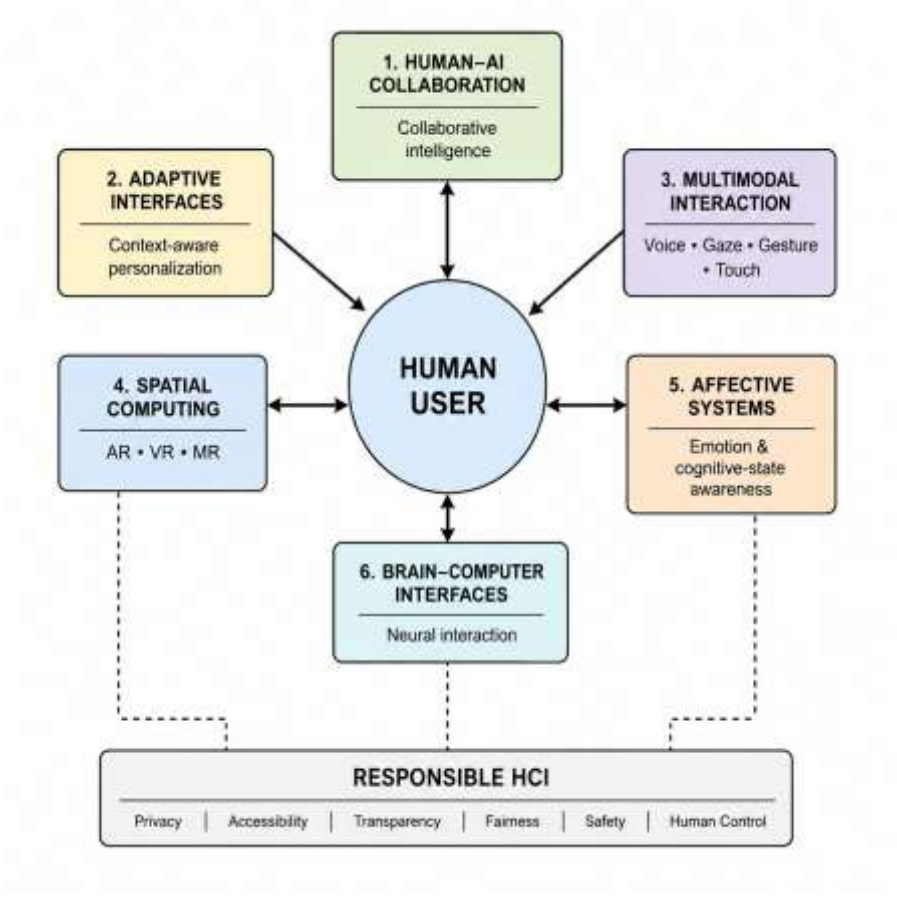


Figure 6. Future human–computer interaction ecosystem integrating human–AI collaboration, adaptive interfaces, multimodal interaction, spatial computing, affective systems, brain–computer interfaces, and responsible design.

Figure 6 conceptualizes the future HCI ecosystem with the human user positioned at the center. Surrounding technologies provide different forms of interaction and computational assistance, while responsible-design principles act as a foundation across the ecosystem.

Human–AI collaboration can support problem solving and decision making. Adaptive systems can personalize interaction according to context. Multimodal interfaces can provide flexible communication channels. Spatial technologies can integrate digital information with physical environments. Affective and neural interfaces can potentially extend interaction toward physiological and cognitive signals.

However, all of these developments depend on cross-cutting principles including privacy, accessibility, transparency, fairness, safety, appropriate trust, and meaningful human control.

The future of HCI should therefore not be evaluated according to the number of intelligent technologies incorporated into an interface. Technological sophistication is valuable only when it produces interaction that genuinely supports human goals.

6.9. Conclusion

Human–Computer Interaction has undergone a substantial transformation from command-oriented computing toward increasingly natural, intelligent, adaptive, and collaborative technological environments.

This chapter has examined that transformation from several complementary perspectives. The foundations of HCI demonstrated the continuing importance of perception, cognition, mental models, feedback, consistency, and human factors. Human-centered design and usability emphasized the importance of iterative development, evaluation, accessibility, and user experience. Emerging technologies demonstrated how AI, generative systems, multimodal interfaces, and immersive environments are expanding the boundaries of interaction. Ethical analysis further demonstrated that usability alone is insufficient when systems influence decisions, collect sensitive information, or operate with increasing autonomy.

The central challenge facing contemporary HCI is therefore no longer simply how to make computers easier to use.

It is how to design increasingly capable computational systems while ensuring that human understanding, judgment, values, and control remain central to interaction.

Future HCI will likely involve systems that can predict, generate, adapt, communicate naturally, sense context, and collaborate with humans. Yet greater technological autonomy does not necessarily imply better interaction. Systems must remain understandable, controllable, inclusive, and aligned with human goals.

Recent research on the future orientation of HCI similarly cautions against purely technology-centered visions and calls for greater attention to human experience, uncertainty, social implications, and alternative futures (Sanchez et al., 2025).

Accordingly, the long-term success of HCI will depend less on replacing human activity with technology and more on designing productive relationships between human and computational intelligence.

The future of interaction is therefore best understood not as human versus computer, but as the continuing development of responsible, adaptive, and mutually supportive human–computer partnerships.

REFERENCES

Amershi, S., Weld, D., Vorvoreanu, M., Fourney, A., Nushi, B., Collisson, P., Suh, J., Iqbal, S., Bennett, P. N., Inkpen, K., Teevan, J., Kikin-Gil, R., & Horvitz, E. (2019). Guidelines for human-AI interaction. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 1–13. https://doi.org/10.1145/3290605.3300233

Arvas, M. M., & Çınar, A. (2026). Detection of underground objects from GPR data using a lightweight YOLO-based approach. *Scientific Reports*.

Capel, T., & Brereton, M. (2023). What is human-centered about human-centered AI? A map of the research landscape. *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, Article 359, 1–23. https://doi.org/10.1145/3544548.3580959

Caragay, E., Xiong, K., Zong, J., & Jackson, D. (2024). Beyond dark patterns: A concept-based framework for ethical software design. *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, Article 291, 1–16. https://doi.org/10.1145/3613904.3642781

Chen, Y., Peng, Y., Tang, J., Camilleri, T. A., Camilleri, K. P., Kong, W., & Cichocki, A. (2025). EEG-based affective brain–computer interfaces: Recent advancements and future challenges. *Journal of Neural Engineering, 22*(3), 031004. https://doi.org/10.1088/1741-2552/ade290

Gulati, S., McDonagh, J., Sousa, S., & Lamas, D. (2024). Trust models and theories in human–computer interaction: A systematic literature review. *Computers in Human Behavior Reports, 16*, 100495. https://doi.org/10.1016/j.chbr.2024.100495
International Organization for Standardization. (2019). *ISO 9241-210:2019 Ergonomics of human-system interaction—Part 210: Human-centred design for interactive systems*. ISO.
Kosch, T., Karolus, J., Zagermann, J., Reiterer, H., Schmidt, A., & Woźniak, P. W. (2023). A survey on measuring cognitive workload in human–computer interaction. *ACM Computing Surveys, 55*(13s), Article 283, 1–39. https://doi.org/10.1145/3582272
Norman, D. A. (2013). *The design of everyday things: Revised and expanded edition*. Basic Books.
Raees, M., Meijerink, I., Lykourentzou, I., Khan, V.-J., & Papangelis, K. (2024). From explainable to interactive AI: A literature review on current trends in human-AI interaction. *International Journal of Human-Computer Studies, 189*, 103301. https://doi.org/10.1016/j.ijhcs.2024.103301
Sanchez, C., Wang, S., Savolainen, K., Epp, F. A., & Salovaara, A. (2025). Let's talk futures: A literature review of HCI's future orientation. *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, Article 487, 1–36. https://doi.org/10.1145/3706598.3713759
Seini, A. B., Adam, I. O., & Preko, M. (2026). Human-AI collaboration in information systems research: A systematic literature review and future research directions. *Human-Intelligent Systems Integration*. https://doi.org/10.1007/s42454-026-00100-7
Shin, Y. (2025). Toward human-centered artificial intelligence for users' digital well-being: Systematic review, synthesis, and future directions. *JMIR Human Factors, 12*, e69533. https://doi.org/10.2196/69533
Shneiderman, B., Plaisant, C., Cohen, M., Jacobs, S., Elmqvist, N., & Diakopoulos, N. (2016). *Designing the user interface: Strategies for effective human-computer interaction* (6th ed.). Pearson.
Tankelevitch, L., Kewenig, V., Simkute, A., Scott, A. E., Sarkar, A., Sellen, A., & Rintel, S. (2024). The metacognitive demands and opportunities of generative AI. *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, Article 680, 1–24. https://doi.org/10.1145/3613904.3642902
Thushara, B., Venugopalan, A., & Sreekanth, N. S. (2026). Multimodal human–computer interaction: A panoptic view. *Engineering Applications of Artificial Intelligence, 179*, 115233. https://doi.org/10.1016/j.engappai.2026.115233
World Wide Web Consortium. (2024). *Web Content Accessibility Guidelines (WCAG) 2.2*. W3C.