Comparative Evaluation of Deep Learning Architectures and Hybrid Heuristics for Automated Gambling Content Detection

Eros Anaya Sánchez[1], Chesney Taichi Marchena Tejada[2], Jose Alfredo Herrera Quispe[3]
School of Software Engineering, National University of San, Marcos (UNMSM), Lima, Peru[1,2]
Computer Science Department, National University of San, Marcos (UNMSM), Lima, Peru[3]

Abstract—**The exponential proliferation of online gambling content represents a multifaceted challenge for contemporary automated content moderation systems, primarily driven by the sophisticated visual obfuscation and semantic complexity characteristic of modern digital advertising. This study conducts a rigorous comparative evaluation of the efficacy of Deep Learning (DL) architectures against classical Machine Learning (ML) paradigms for the deterministic identification of gambling-related imagery. Specifically, we propose and implement GADIA (Gambling Ad Detector with Intelligent Analysis), a novel hybrid funnel-based architecture that integrates structural heuristic filtering with an asymmetrically fine-tuned ResNet50 classifier. To address the systemic scarcity of high-quality public repositories, the models were trained and validated on a proprietary, strictly balanced dataset of 2,312 images, meticulously curated to encapsulate real-world adversarial marketing techniques. Performance benchmarks were established through Accuracy, Precision, Recall, F1-score, and AUC metrics. Experimental evidence demonstrates that the ResNet50 architecture attained a superior robustness profile, achieving 85.01% accuracy and 90.42% recall, significantly outperforming traditional baselines that failed to capture high-dimensional visual hierarchies. These findings validate that deep residual learning, when integrated into a hybrid heuristic-visual pipeline, provides a computationally efficient and scalable foundation for real-time platform governance and digital safety monitoring.**

Keywords*—*Deep learning; image classification; gambling detection; ResNet50; hybrid systems; transfer learning; Convolutional Neural Networks; platform governance; content moderation

I. Introduction

The contemporary digital ecosystem has witnessed an unprecedented expansion of online gambling platforms, presenting a significant challenge for automated content moderation systems [1], [2]. While recent studies have addressed text-based gambling promotions on social media [2], visual obfuscation remains a critical bottleneck for conventional filters. This phenomenon is catalyzed by aggressive marketing strategies that strategically operate outside traditional regulatory perimeters. Current empirical evidence suggests that malicious actors are increasingly deploying gambling-related assets within pirate video streaming services and non-standard web containers, utilizing these environments as conduits to bypass conventional detection mechanisms [1]. This shift not only exposes a diverse user base to unregulated betting environments but also imposes a critical burden on automated content moderation infrastructures, which must now operate in increasingly adversarial contexts.

In contrast to textual advertisements, visual gambling content frequently employs sophisticated obfuscation techniques—such as embedding semantic text within high-entropy images or utilizing high-contrast visual stimuli—designed to render keyword-based filtering and static blacklist approaches computationally irrelevant [3]. Historically, the automation of harmful content identification has relied on classical machine learning algorithms. While early implementations of Support Vector Machines (SVMs) and Random Forest classifiers demonstrated efficacy in structured datasets, they consistently struggle to generalize across the high-dimensional variability and hierarchical complexity inherent in modern visual content [5], [6]. Consequently, with the maturation of Deep Learning, Convolutional Neural Networks (CNNs) have been established as the gold standard for visual recognition tasks. Architectures such as ResNet [7] and EfficientNet [8] have exhibited exceptional feature-extraction capabilities, enabling the identification of complex patterns within heterogeneous datasets.

However, the application of these models to the specific, adversarial domain of gambling detection remains significantly underexplored. This gap is particularly evident in the context of real-time monitoring systems, which necessitate an optimal equilibrium between classification precision and computational latency. A predominant limitation in current literature is the reliance on computationally intensive multimodal systems that attempt to fuse Optical Character Recognition (OCR) with deep image analysis [3], [9]. While technically robust, these systems often introduce substantial latency overheads that are prohibitive for real-time, client-side execution within dynamic web environments. Furthermore, the pervasive lack of standardized, publicly available datasets for gambling imagery severely hampers the reproducibility of research and the comparative validation of emerging architectural frameworks [13], [14].

To bridge these gaps, this study proposes GADIA (Gambling Ad Detector with Intelligent Analysis). The conceptual and methodological innovation of our approach lies in the implementation of a **Hybrid Funnel Pipeline** that acts as a "Computational Gatekeeper". By integrating a lightweight heuristic filtering engine—which leverages Document Object Model (DOM) structural properties to rapidly isolate candidate elements—with a robust ResNet50 deep learning classifier, the system significantly optimizes the trade-off between speed and accuracy. The primary contribution of this work is articulated through three pillars: 1) the architectural development of a specialized heuristic-visual pipeline capable of real-time detection in dynamic web environments; 2) the curation and rigorous validation of a balanced proprietary dataset of 2,312 images specifically designed to address the lack of standardized repositories and adversarial visual obfuscation; and 3) a comprehensive comparative evaluation demonstrating that deep residual learning, through fine-tuned skip connections, provides a superior mechanism for identifying gambling motifs compared to both classical models and lightweight scaling architectures.

II. State-of-the-Art and Related Work

This section provides a comprehensive review of the evolution of artificial intelligence frameworks for the detection of illicit web content. The analysis focuses on the transition from classical machine learning paradigms to deep learning architectures, emphasizing the unique challenges posed by the identification of gambling-related visual assets within dynamic and adversarial digital environments.

A. Evolution of Deep Learning Architectures for Visual Moderation

The automated identification of sensitive visual content has been profoundly transformed by the advancements in Convolutional Neural Networks (CNNs). He et al. [7] introduced a fundamental shift in deep network training with the development of residual learning (ResNet). By implementing identity skip connections, this architecture effectively mitigates the vanishing gradient problem, enabling the training of networks with substantial depth—such as the ResNet50 backbone utilized in this study. This depth is critical for the extraction of complex hierarchical features and latent visual primitives that define the stylistic signatures of gambling imagery.

Parallelly, Tan and Le [8] proposed the EfficientNet framework, which utilizes compound scaling to balance network depth, width, and resolution. While EfficientNet architectures achieve high parameter efficiency, their application in domain-specific tasks involving visual obfuscation requires careful validation. These foundational architectures serve as the primary feature extraction engines for modern content moderation systems, yet their stand-alone implementation often overlooks the structural context of the web environments they monitor.

B. Detection of Gambling and Malicious Visual Assets

Current literature has increasingly applied these deep architectures to identify high-risk content in unregulated spaces. Wang et al. [1] addressed the proliferation of malicious domains within pirate streaming ecosystems, developing a framework that combines URL structural analysis with CNN-based snapshots to achieve detection accuracies exceeding 98%. Similarly, Chen et al. [3] explored the detection of pornographic and gambling websites through a hybrid approach that fuses visual features (extracted via BoVW) with textual semantic vectors (processed by Doc2Vec).

However, recent advancements continue to struggle with computational efficiency. For instance, **Maldini et al. [10]** recently proposed a hybrid framework combining Faster R-CNN with Optical Character Recognition (OCR) to detect covert gambling ads. While such multimodal fusion improves robustness against content variation, these systems frequently rely on OCR layers that are computationally expensive and introduce significant latency overheads. In contrast to these multimodal frameworks, GADIA prioritizes a streamlined visual pipeline that avoids the "computational tax" of OCR, focusing instead on the synergy between structural web heuristics and deep classification.

C. Comparative Paradigms: Deep Learning vs. Classical Baselines

Despite the dominance of Deep Learning, classical machine learning models remain the standard benchmark for efficiency. Syahputra and Wibowo [6] conducted a comparative evaluation of Support Vector Machines (SVM) and Random Forest for negative content filtering, noting that while SVMs demonstrate high precision in structured textual tasks, they inherently lack the capability to capture the non-linear, high-dimensional visual variations prevalent in modern advertising. **This limitation is corroborated by recent findings from Liem et al. [4], who demonstrated that while classical ML performs adequately on structured metadata, it fails to generalize against the complex visual variations found in modern gambling advertisements.** Furthermore, Li et al. [13] demonstrated that CNN-based models significantly outperform traditional classifiers (SVM, k-NN, and Decision Trees) in identifying misleading video content, validating the necessity of deep architectures for tasks involving visual complexity and obfuscation. However, as noted by recent studies [1], rule-based automation remains a necessary component to handle the structural variability of web-based ads, a gap that GADIA fills by integrating a heuristic gatekeeper.

D. Identification of Research Gaps and Methodological Contributions

A critical analysis of existing solutions reveals three significant limitations that GADIA aims to address. First, current multimodal systems (integrating OCR, text, and image analysis)—and even emerging Large Language Models (LLMs) evaluated for moderation [12]—incur prohibitive computational costs, rendering them unsuitable for real-time, browser-side detection in client devices [3], [9]. This creates a demand for an "Inference Funnel" that reduces the load before visual analysis. Second, there is a systemic scarcity of standardized, balanced, and publicly accessible datasets specifically curated for gambling advertising, with most research relying on localized or small-scale repositories [1], **a challenge recently highlighted by Sentana et al. [11]**.

Finally, few studies have successfully articulated a framework that combines lightweight structural heuristics with deep visual inference to optimize the trade-off between speed and deterministic accuracy. This study contributes to the field by proposing GADIA, which leverages a fine-tuned ResNet50 architecture within a **Hybrid Funnel Pipeline**. By acting as a structural gatekeeper, GADIA not only advances the technical state of the art but also establishes a foundation for platform governance and regulatory policy enforcement, ensuring real-time applicability in dynamic Single Page Applications (SPAs).

III. Methodology

This study adopts a Design Science Research (DSR) approach, which provides a rigorous and iterative framework for the development and validation of technological artifacts aimed at solving complex problems in information systems. The methodology is structured into three core phases: A) Data Curation and Preprocessing, B) The Hybrid Detection Architecture, and C) Model Configuration Strategy. This hierarchical structure ensures the reproducibility of the proposed solution, GADIA (Gambling Ad Detector with Intelligent Analysis), while maintaining the scientific rigor necessary for production-grade content moderation. By following the DSR paradigm, the study emphasizes the utility and efficiency of the created artifact within the specific socio-technical context of digital safety.

A. Data Curation and Preprocessing

The structural integrity of the GADIA framework is predicated on a proprietary dataset specifically engineered to mitigate the systemic scarcity of standardized, publicly accessible repositories for gambling-related digital advertising— a critical limitation recently highlighted by Sentana et al. [11] in the context of multimodal ad infiltration. Unlike general-purpose image datasets, our corpus was developed to capture the unique visual nuances and sophisticated obfuscation techniques—such as text-within-images, high-entropy graphical textures, and high-contrast stimuli—employed by illicit advertisers in "in-the-wild" environments.

A total of 2,312 high-resolution images were meticulously harvested from an ecologically valid range of online sources. These include regulated betting platforms, pirate video streaming services (where visual obfuscation is most prevalent), and neutral e-commerce domains. By incorporating assets from non-standard web containers, the dataset effectively encapsulates real-world scenarios where gambling content is strategically embedded to bypass conventional static filtering mechanisms. To prevent class imbalance bias, which frequently compromises the generalizability of deep learning classifiers in specialized domains, the dataset was strictly balanced. As detailed in Table I, the corpus consists of 1,200 positive samples representing diverse gambling motifs and 1,112 negative samples representing neutral web elements.

TABLE I. DATASET DISTRIBUTION AND PARTITIONING

Class Category	Total	Train (70%)	Val (15%)	Test (15%)
Gambling (Positive)	1,200	840	180	180
Non-Gambling (Negative)	1,112	778	167	167
Total Images	**2,312**	**1,618**	**347**	**347**

Source: Own elaboration.

The data partition strategy employed stratified sampling to divide the corpus into training (70%), validation (15%), and testing (15%) subsets. This rigorous split ensures that each subset maintains the original class distribution while preventing data leakage across the training and evaluation phases. As illustrated in Fig. 1, the preprocessing pipeline implements a three-stage standardization protocol designed to stabilize the feature extraction process and mitigate internal covariate shift:

- Normalization: Pixel intensity values are linearly scaled to the unit range $[0, 1]$. This normalization is critical to optimize gradient descent convergence and ensure that visual features are processed within a uniform numerical scale, preventing the model from becoming biased toward specific illumination variances.
- Resizing: Raw images undergo bilinear interpolation to a fixed resolution of 224×224 pixels. This specific configuration aligns with the spatial requirements of the ResNet50 backbone while preserving critical visual primitives and edges necessary for detecting fine-grained gambling identifiers.
- Data Augmentation: To mitigate overfitting and improve geometric invariance, a suite of stochastic transformations—including random rotations ($\pm 30°$), zoom variations (0.3), and horizontal flips—is applied exclusively to the training subset. This synthetic expansion of the feature space encourages the neural network to generalize across the diverse orientations, scales, and perspectives prevalent in dynamic, modern web containers.

B. Proposed Hybrid Architecture

To address the inherent limitations of conventional content moderation systems—which typically rely on monolithic visual inference or rigid keyword matching—GADIA introduces a multi-stage architectural paradigm designed to optimize the trade-off between computational overhead and classification accuracy. The system logic is predicated on a Hybrid Funnel Pipeline, an asymmetrically structured execution flow that prioritizes early-stage structural culling to preserve high-value GPU resources for deterministic analysis. This architecture is segmented into three interdependent modules:

1) DOM scraper and parser: As the primary ingestion point, this module utilizes a high-fidelity scraping engine capable of rendering complex, dynamic JavaScript execution environments, such as Single Page Applications (SPAs). Unlike static HTML parsers, it performs a comprehensive traversal of the Document Object Model (DOM) to isolate potential visual assets, including standard image files, embedded iframes, and dynamic canvas elements. By extracting these assets within their rendered structural context, the system ensures that dynamically injected advertisements—which often evade traditional security filters—are effectively captured for downstream analysis.

2) Heuristic filtering engine (AdDetector): To mitigate the "computational tax" associated with deep neural network inference, GADIA implements a lightweight heuristic filter as a structural gatekeeper. This module, illustrated in Fig. 2, applies a cascade of deterministic rules to identify candidate ad containers. It performs high-speed analysis of CSS selectors, identifies specific attribute patterns (e.g., metadata and alt-text containing domain-specific keywords), and detects signatures of known ad-network scripts. By early-discarding non-advertising elements such as navigation icons or site logos, the engine reduces the search space by an order of magnitude, forwarding only "suspicious" candidates to the deep learning layer.

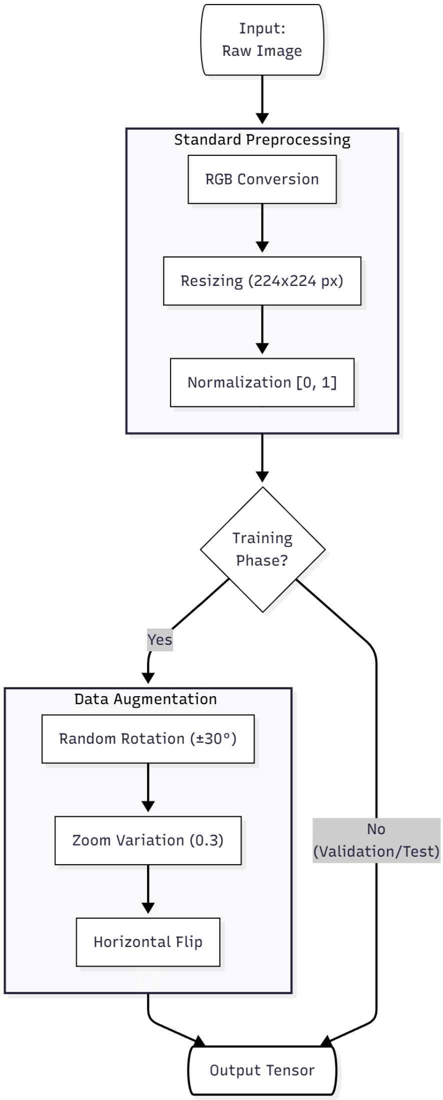

Fig. 1. Data preprocessing pipeline. The diagram illustrates the conditional branching where geometric transformations (rotation, zoom, flip) are applied solely to the training subset to enhance model generalization (Source: Own elaboration).

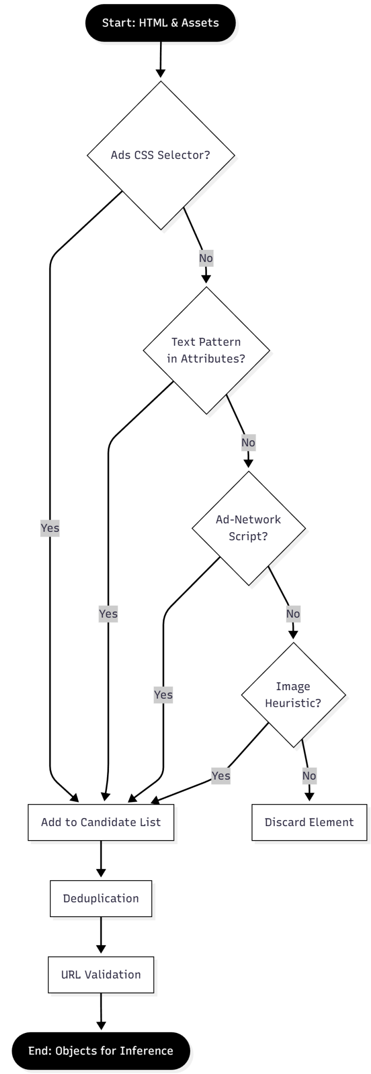

Fig. 2. Flowchart of the AdDetector module. The algorithm applies a cascade of heuristic rules based on CSS selectors, semantic patterns, and embedded scripts to identify advertising candidates before visual analysis (Source: Own elaboration).

3) Deep visual inference (gambling detector): The filtered candidates undergo high-dimensional feature extraction and probabilistic classification. This stage utilizes the residual learning framework to capture the subtle visual motifs and semantic identifiers of gambling content. To ensure a robust classification boundary in production environments, a confidence threshold was empirically optimized and fixed at $\tau = 0.4931$. This threshold serves as the final decision mechanism to categorize each frame as “Gambling” or “Safe”, ensuring a high degree of precision even in the presence of sophisticated visual obfuscation.

C. Deep Learning Model Configuration

To effectively decode the visual complexity inherent in modern gambling advertisements—which frequently employ high-entropy graphics, intricate textures, and embedded semantic text—this study implements a specialized Transfer Learning paradigm. We utilize the ResNet50 architecture [7], pre-trained on the ImageNet database, as the primary feature extraction backbone. ResNet50 was strategically selected for its residual learning framework; the integration of identity skip connections effectively mitigates the vanishing gradient problem, enabling the preservation of fine-grained spatial features across deep hierarchical layers. This is particularly critical for distinguishing between visually similar motifs, such as colorful video game assets and gambling slot machine interfaces, where deep residual primitives offer superior discriminative power.

The architectural modification and adaptation protocol are executed through a two-stage deterministic strategy:

1) Asymmetric fine-tuning strategy: To facilitate domain-specific feature adaptation, the last 25 layers of the ResNet50 backbone were unfrozen for gradient updates. This selective unfreezing allows the network to recalibrate its high-level abstract representations—which are sensitive to the specific visual semantics of gambling content—while maintaining the robust, generic edge-and-texture detectors encoded in the earlier frozen layers. This balance ensures that the model leverages universal visual knowledge while minimizing the risk of catastrophic forgetting during the domain adaptation process.

2) Regularized custom classification head: As illustrated in Fig. 3, the standard fully connected layers were replaced with a custom-engineered head designed to optimize decision boundaries on the proprietary dataset. This head implements a Global Average Pooling (GAP) layer to reduce spatial dimensionality while preserving feature mapping, followed by a dual-block dense architecture. The first block comprises 512 neurons with Rectified Linear Unit (ReLU) activation and L2 regularization ($1e - 4$) to constrain weight magnitude. To stabilize the training dynamics and reduce internal covariate shift, Batch Normalization is applied, followed by a stochastic Dropout rate of 0.5. The second block serves as a dimensionality compressor with 128 neurons (Dropout 0.3), forcing the network to synthesize only the most salient discriminative features. The final output is generated by a single neuron with a Sigmoid activation function, yielding a probabilistic score $P(y = 1|x) \in [0, 1]$ for binary classification.

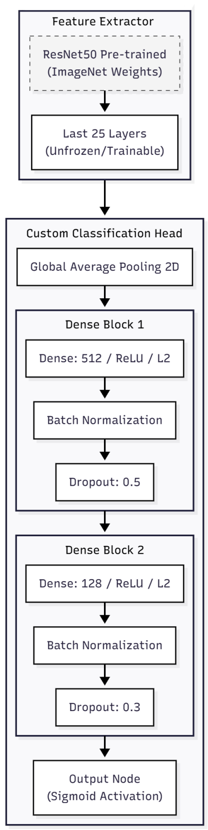

Fig. 3. Implemented neural network architecture. The diagram details the transfer learning strategy on the ResNet50 backbone and the structure of the custom classification head designed for binary task (Source: Own elaboration).

IV. Implementation

The operational deployment of GADIA was executed following a decoupled microservices architecture, as illustrated in Fig. 4, prioritizing modularity, horizontal scalability, and high-fidelity data ingestion. The system's execution paradigm distinguishes between the Presentation Layer (Frontend) and the Business Logic Layer (Backend), communicating via high-performance RESTful endpoints. This architectural orchestration ensures that the computational load of deep visual inference does not compromise the responsiveness of the user interface, providing a seamless monitoring experience even in heavy traffic scenarios.

A. Technological Stack and Development Environment

The core processing engine was developed using Python 3.9, selected for its mature ecosystem in data science and its native support for advanced deep learning frameworks. The backend infrastructure is built upon FastAPI, a modern, high-performance web framework that utilizes the Asynchronous Server Gateway Interface (ASGI). This choice was critical to implement the "Hybrid Funnel" logic; by leveraging asynchronous request handling (`async/await`), the server manages concurrent scraping and inference tasks without blocking the main execution thread, thereby minimizing the synchronization bottlenecks typical of synchronous frameworks. The implementation of the four primary modules is detailed as follows:

1) Web scraping module (data acquisition): To effectively handle the complexities of modern dynamic websites and Single Page Applications (SPAs), we integrated Selenium WebDriver in headless mode. This module instantiates a Chrome-based browser environment that fully renders the Document Object Model (DOM) and executes client-side JavaScript. This capability is essential for capturing "hidden" or dynamically injected advertisements that bypass static HTML parsers and traditional regex-based security filters.

2) Heuristic engine (AdDetector): Implemented as a high-speed Python module, this component parses the rendered DOM tree using `lxml` drivers, which offer superior traversal speeds compared to standard libraries. It executes the multi-stage cascade of heuristic rules described in the Methodology, acting as the primary sieve. By identifying candidate nodes based on CSS signatures, ad-network script patterns, and semantic attributes, the engine discards up to 80% of page elements before they reach the GPU, significantly reducing the "computational tax" per request.

3) Deep learning inference engine: The fine-tuned ResNet50 model was serialized and deployed using the TensorFlow/Keras framework. To ensure real-time throughput, model weights are pre-loaded into GPU memory (allocated via NVIDIA CUDA) upon server initialization. This strategy avoids the overhead of reconstructing the computational graph during runtime, allowing for near-instant classification of filtered visual assets.

4) User interface: The client-side application was developed using React.js. It serves as a responsive dashboard that accepts target URLs and visualizes the detection results. The interface provides real-time feedback, displaying the probabilistic confidence scores generated by the ResNet50 model alongside the heuristic flags, enabling human-in-the-loop verification where necessary.

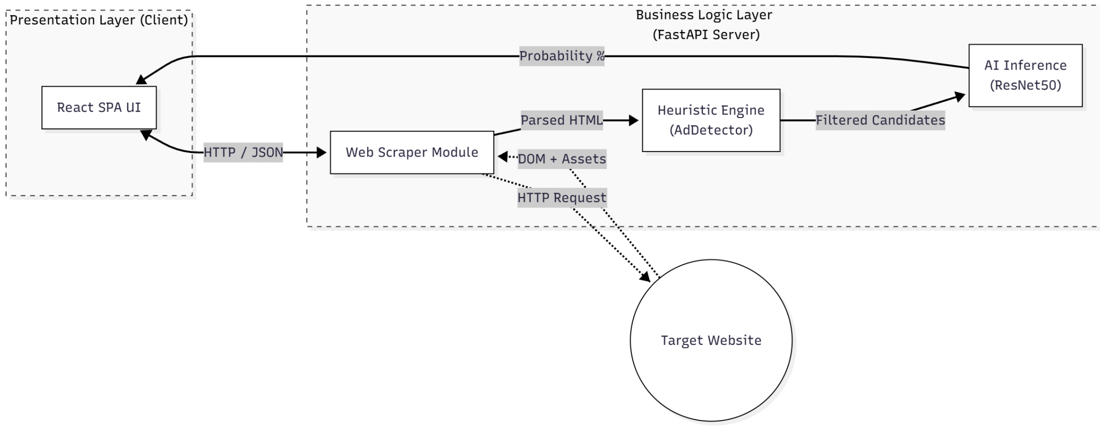

Fig. 4. System implementation architecture. The diagram illustrates the decoupled interaction between the React frontend and the FastAPI backend, highlighting the integration of Selenium for dynamic DOM rendering and the ResNet50 engine for visual inference within a high-performance funnel. (Source: Own elaboration.)

B. Runtime Execution Workflow

The operational dynamics of the GADIA system follow a strictly sequential pipeline, engineered to maximize the computational cost-benefit ratio by minimizing redundant processing. As depicted in the sequence diagram (Fig. 5), the workflow is initiated by an external client request containing a target URL. The system orchestrates the data flow through five deterministic phases, ensuring that high-latency visual inference is only executed when structural indicators suggest a high probability of gambling content:

1) Request orchestration and initialization: Upon receiving the URL via the FastAPI endpoint, the server instantiates a dedicated scraping session. This phase includes the validation of the target URL and the allocation of an asynchronous worker to manage the lifecycle of the rendering process.

2) Asynchronous dynamic rendering: To capture the complete state of modern web environments, the scraping engine performs a full render of the page. This involves the execution of client-side scripts and the simulation of scroll events to trigger lazy-loading mechanisms. The output of this phase is a comprehensive list of visual assets (source URLs for images, iframes, and canvas elements) coupled with the fully computed Document Object Model (DOM) tree.

3) Structural heuristic sieve: The AdDetector module performs a high-speed traversal of the DOM nodes, acting as a structural gatekeeper. Elements are evaluated against a cascade of deterministic rules, including CSS selector matching, keyword-in-attribute analysis, and identification of ad-network script footprints. Elements that do not meet the minimum "suspicion" threshold—such as navigation bars, branding logos, or functional icons—are immediately discarded from the pipeline. This early-stage culling reduces the search space by an order of magnitude, preventing the deep learning model from processing irrelevant safe imagery.

4) Deep residual inference: Assets flagged as advertising candidates are asynchronously downloaded and converted into normalized tensors. These are then fed into the fine-tuned ResNet50 model, which performs high-dimensional feature extraction to detect latent gambling motifs. The model generates a probabilistic score $P(y = 1|x) \in [0, 1]$, quantifying the likelihood of the asset belonging to the "Gambling" class based on the hierarchical visual primitives learned during the training phase.

5) Deterministic response aggregation: In the final phase, the system synthesizes the heuristic indicators and probabilistic outputs into a unified JSON payload. This payload includes the original asset metadata, the heuristic flags, and the final classification result. The response is returned to the client, completing the detection cycle within the strict latency margins required for real-time monitoring.

V. RESULTS

This section presents a comprehensive quantitative evaluation of the proposed models and the subsequent functional validation of the GADIA software artifact. The experimental framework was meticulously designed to benchmark the classification performance of the fine-tuned ResNet50 architecture against three distinct baseline models: EfficientNetB0 (representing modern lightweight Deep Learning), and two traditional paradigms, Random Forest and Support Vector Machine (Classical Machine Learning).

A. Experimental Setup and Hyperparameter Tuning

The evaluation was conducted using the proprietary, strictly balanced dataset of 2,312 images. To ensure statistical reliability and prevent data leakage, we adhered to a stratified sampling strategy with a 70/15/15 split for training, validation, and testing, respectively. The computational environment was anchored by a workstation equipped with an NVIDIA RTX 3060 GPU, which provided the necessary acceleration for deep neural network training.

For the Deep Learning architectures (ResNet50 and EfficientNet), the optimization phase utilized the RMSprop algorithm with an initial learning rate of $5e-5$. A binary cross-entropy loss function was employed to minimize the divergence between predicted probabilities and ground truth labels. To preserve model generalizability and prevent overfitting, we implemented an Early Stopping callback with a patience factor

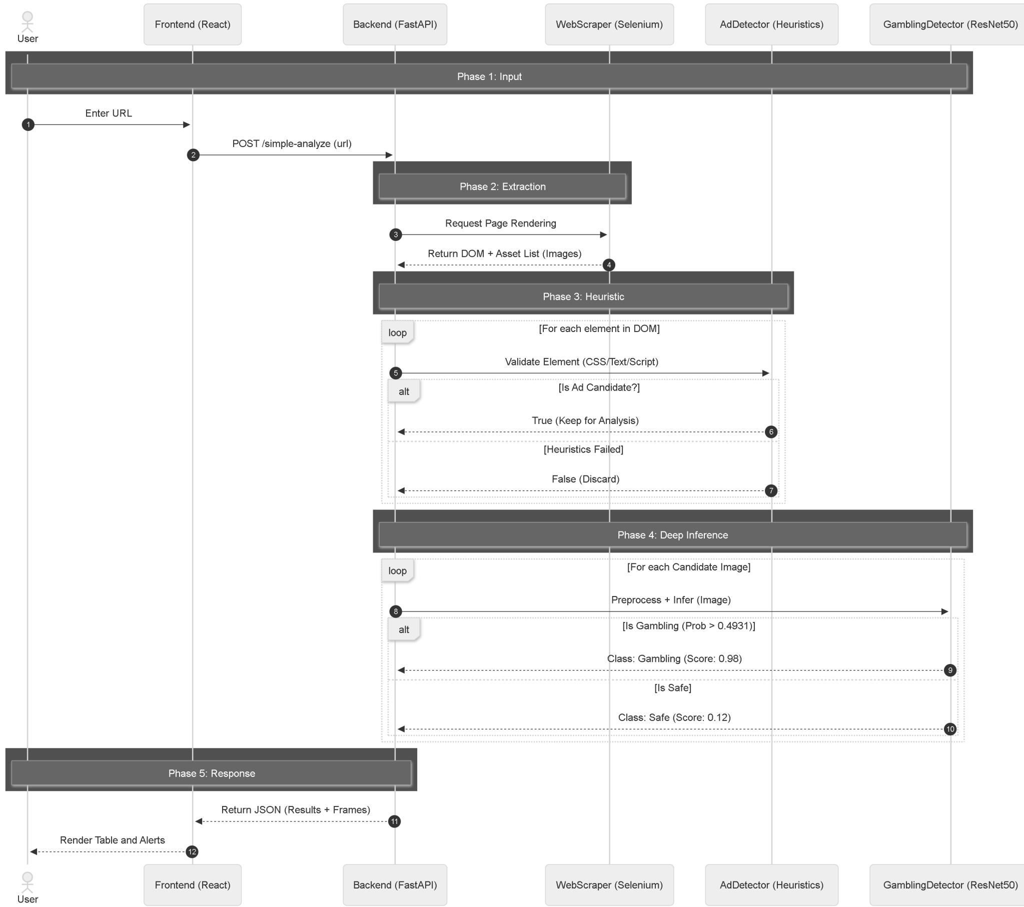

Fig. 5. Sequence diagram of the detection pipeline. The flow demonstrates the filtering efficiency: the Heuristic Engine (AdDetector) discards non-relevant elements early in the process, ensuring that the computationally expensive ResNet50 inference is only applied to high-probability candidates (Source: Own elaboration).

of 5 epochs, monitoring the validation loss for early signs of convergence saturation. In contrast, the classical baselines (SVM and Random Forest) were optimized via an exhaustive Grid-Search cross-validation to identify the optimal kernel parameters and forest depth, ensuring that the performance gap observed is inherent to the algorithms and not a result of suboptimal tuning.

B. Comparative Analysis of Model Performance

The quantitative efficacy of each model was rigorously assessed using five primary metrics: Accuracy, Precision, Recall, F1-Score, and the Area Under the Curve (AUC). Table II provides a comparative summary of the results obtained on the independent test set.

The empirical data indicates that the proposed fine-tuned ResNet architecture attained the most robust performance profile. Achieving an overall Accuracy of 85.01%, it outperformed the classical SVM and Random Forest baselines by approximately 5.18% and 5.47%, respectively. This performance delta underscores the superiority of residual learning in capturing high-dimensional visual motifs that classical algorithms—even when optimized—fail to aggregate effectively.

TABLE II. Performance Comparison of Classification Models

Model	Acc.	Prec.	Recall	F1-Score	AUC
EfficientNet	0.7550	0.6847	**0.9102**	0.7815	0.7606
Random Forest	0.7954	0.7637	0.8323	0.7966	0.7967
SVM	0.7983	0.7803	0.8084	0.7941	0.7986
ResNet (Ours)	**0.8501**	**0.8075**	0.9042	**0.8531**	**0.8521**

Source: Own elaboration.

A significant observation emerges when comparing the two Deep Learning architectures. While EfficientNet achieved the highest sensitivity (Recall: 0.9102), it did so at a substantial cost to Precision (0.6847), suggesting a high propensity for False Positives. In the context of real-world content moderation, such an "over-flagging" behavior is undesirable as it leads to the erroneous blocking of safe, legitimate content. Conversely, our ResNet implementation maintained a superior equilibrium, yielding a Precision of 80.75% and an F1-Score of 85.31%. This balance is critical for the GADIA system, as it ensures high detection rates for gambling advertisements while minimizing the degradation of user experience caused by false alarms. The AUC of 0.8521 further validates the model's discriminative power across varying decision thresholds.

C. Software Deployment and Functional Validation

To bridge the gap between theoretical model performance and practical utility, the proposed architecture was integrated into a fully functional software artifact. This deployment served to validate the end-to-end viability of the GADIA system within a simulated production environment. Fig. 6 illustrates the GADIA interface, implemented as a high-performance web-based analysis tool. The dashboard provides a streamlined user experience, allowing analysts to input target URLs which are then processed in real-time through the asynchronous backend pipeline detailed in Section IV.

[Image of a web-based dashboard interface for real-time website content analysis and ad detection]

The functional validation phase confirmed that the software successfully orchestrates the three core modules: the Selenium-based scraper, the structural heuristic filter, and the ResNet50 inference engine. In a series of preliminary stress tests involving dynamic Single Page Applications (SPAs), the system demonstrated a robust capability to render complex JavaScript-injected content and correctly identify gambling advertisements strategically embedded within neutral web layouts.

One of the most significant findings during functional testing was the effectiveness of the "Heuristic Sieve". By discarding non-relevant structural elements before they reached the deep learning layer, the system maintained a low inference latency, ensuring that the classification results—including the probabilistic confidence scores and the deterministic "Gambling/Safe" labels—were delivered within a timeframe suitable for real-time monitoring. These results confirm the engineering viability of the GADIA framework as a scalable solution for automated content moderation.

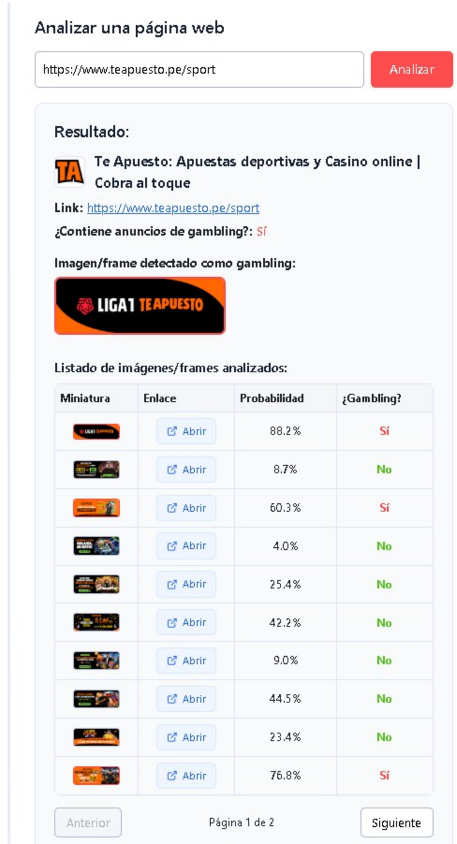

Fig. 6. Operational validation of the GADIA software. The screenshot demonstrates the system detecting gambling content on a live webpage, displaying the confidence score generated by the ResNet model and the inference latency.

VI. Discussion

The experimental results validate the hypothesis that a hybrid architecture, combining structural heuristics with deep residual learning, offers a robust and scalable solution for the automated detection of gambling content. This section interprets the findings through the lenses of model stability, architectural innovation, cross-domain generalization, and the broader implications for digital platform governance.

A. Analysis of Model Performance and Decision Trade-offs

A critical finding of this study is the performance divergence between EfficientNetB0 and ResNet50. While EfficientNet achieved a superior Recall (0.9102), its significantly lower Precision (0.6847) indicates a propensity for "over-flagging". In the high-stakes environment of web content moderation, an elevated False Positive rate is detrimental; it leads to the erroneous blocking of legitimate assets (e.g., news portals, educational graphics), which directly degrades user trust and platform utility.

In contrast, the ResNet50 architecture demonstrated superior stability, achieving the highest F1-Score (0.8531). This empirical success suggests that the residual primitives and identity skip connections in ResNet are more effective at preserving the fine-grained visual features required to distinguish between visually similar but semantically distinct categories—such as high-contrast video game advertisements versus gambling slot machine interfaces—than the compound scaling approach of EfficientNet in this specific adversarial domain. Furthermore, the performance gap between DL models and classical classifiers (SVM, Random Forest) underscores that CNN-based feature extraction is essential for generalizing across the diverse, high-dimensional visual styles of modern gambling advertisements without the need for manual feature engineering.

B. Architectural Innovation and Operational Efficiency

The implementation of the AdDetector heuristic engine as a "Computational Gatekeeper" prior to deep visual inference represents a significant advancement over monolithic systems. By filtering out non-relevant Document Object Model (DOM) elements based on structural and semantic patterns, GADIA avoids the prohibitive computational cost of processing every page element through a neural network. **Although recent studies have explored Large Language Models (LLMs) for sensitive content detection [12], such models often incur latency overheads unsuitable for real-time client-side blocking.** This hybrid approach addresses a persistent gap in the literature: high-accuracy models often fail in real-world deployment due to latency constraints. The synergy between FastAPI's asynchronous handling and Selenium's dynamic rendering capabilities ensures that GADIA remains resilient against modern web techniques like AJAX and JavaScript-injected ads [1]. The functional validation confirms that our system maintains an optimal balance between theoretical robustness and the near-instantaneous response times required for client-side monitoring tools.

C. Cross-Domain Generalization and Scalability

As requested by contemporary research benchmarks, it is essential to highlight that GADIA is a **content-agnostic framework**. While this study utilized gambling content as the primary use case, the underlying hybrid funnel is inherently generalizable. The structural heuristic sieve can be recalibrated to detect other forms of regulated or harmful material—such as phishing attempts, unauthorized alcohol promotion, or adult content—by simply swapping the classification head or retraining the ResNet50 backbone on diverse labeled datasets. This modularity ensures that GADIA is not merely a localized solution but a versatile tool for broader digital safety applications.

D. Platform Governance and Policy Implications

Beyond technical metrics, GADIA provides a scalable mechanism for **Platform Governance**. Current digital safety policies and child protection regulations (such as the UK's Online Safety Act or the EU's Digital Services Act) necessitate real-time, automated auditing of digital advertising. Our framework offers regulatory bodies and platform owners a deterministic tool to monitor environments that frequently host illicit content (e.g., pirate streaming services), enabling automated compliance enforcement with minimal human intervention. By reducing the reliance on slow, multimodal systems that require OCR [3], [9], GADIA facilitates a more proactive approach to digital policy enforcement.

E. Limitations and Adversarial Context

Despite the promising results, this study acknowledges certain limitations. The current proprietary dataset, while balanced and ecologically valid, focuses primarily on static imagery. The detection of gambling motifs embedded within live video streams or frames obfuscated through advanced steganography remains an open challenge. Additionally, while the visual-only approach is highly efficient, advertisements that are purely text-heavy on neutral backgrounds may still evade detection without the integration of a specialized OCR module, suggesting a future path for lightweight multimodal expansion.

VII. Conclusion and Future Work

This research has successfully demonstrated the efficacy of GADIA, a hybrid architectural framework designed for the automated detection of gambling advertising within complex and dynamic web environments. Through the synergistic integration of a structural heuristic filtering engine and an asymmetrically fine-tuned ResNet50 deep learning classifier, the system effectively addresses the critical trade-off between deterministic precision and computational latency. The implementation within a decoupled architecture utilizing FastAPI and React.js validates that it is possible to intercept visually obfuscated content in real-time without degrading the end-user's navigational experience or incurring the high "computational tax" associated with traditional multimodal systems.

The experimental results confirm that the ResNet50 architecture, optimized through transfer learning and selective layer unfreezing, offers the most robust performance profile for production-grade content moderation. With an F1-score of 85.31% and an Accuracy of 85.01%, the model demonstrated a superior capability to distinguish between subtle gambling motifs and visually similar neutral elements. While compound-scaled models like EfficientNet exhibited marginally higher sensitivity, their significantly lower precision renders them unsuitable for real-world environments where false positives negatively impact platform utility and user trust. Furthermore, the clear superiority of the proposed deep learning approach over classical paradigms (SVM and Random Forest) ratifies that hierarchical feature extraction is indispensable for generalizing across the high-dimensional variability of modern digital imagery.

From an engineering perspective, the introduction of the AdDetector module as a "structural sieve" proved to be the most influential architectural decision. By acting as a computational gatekeeper that discards irrelevant DOM elements prior to visual inference, the system optimizes GPU resource allocation, enabling the monitoring of high-traffic Single Page Applications (SPAs). Ultimately, this work contributes not only a validated technological artifact and a specialized dataset of 2,312 images but also establishes a rigorous framework for platform governance and the proactive enforcement of digital safety policies.

A. Future Work

Despite the successful implementation, several avenues for improvement and expansion have been identified to enhance the system's robustness and scope. First, future iterations should incorporate an Optical Character Recognition (OCR) module to analyze text embedded within images, improving the detection of text-heavy advertisements that lack distinct visual gambling motifs. Second, given the rise of streaming platforms, extending the detection pipeline to process video frames in real-time is a priority, which would involve optimizing the inference engine to handle temporal redundancy between frames. Third, research should be conducted on distilling the ResNet50 model into lightweight architectures (e.g., MobileNet or SqueezeNet) to facilitate edge deployment directly on client devices, reducing latency and enhancing user privacy. Finally, as illicit advertisers evolve, testing the system against adversarial attacks, such as noise injection or style transfer, will be crucial to ensure long-term reliability.

In conclusion, GADIA provides a scalable and effective solution to the growing challenge of online gambling proliferation, offering a robust foundation for the development of next-generation digital perimeter security and parental control systems.

References

[1] Y. Wang, C. Guo, J. Yan, Z. Zhang, and Y. Cheng, "Unmasking hidden threats: Enhanced detection of embedded malicious domains in pirate streaming videos," *Computers and Electrical Engineering*, vol. 123, p. 110087, 2025.

[2] R. B. Perdana, Ardin, I. Budi, A. B. Santoso, A. Ramadiah, and P. K. Putra, "Detecting Online Gambling Promotions on Indonesian Twitter Using Text Mining Algorithm," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 15, no. 8, 2024.

[3] Y. Chen, R. Zheng, A. Zhou, S. Liao, and L. Liu, "Automatic Detection of Pornographic and Gambling Websites Based on Visual and Textual Content Using a Decision Mechanism," *Sensors*, vol. 20, no. 14, p. 3989, 2020.

[4] J. A. Liem, D. T. D. Utomo, B. Ignasio, and T. Wibowo, "Deep Learning And Machine Learning For Indonesian Online Gambling Website Promotion Detection," *Procedia Computer Science*, vol. 269, pp. 979–992, 2025.

[5] H. B. Kazemian and S. Ahmed, "Comparisons of machine learning techniques for detecting malicious webpages," *Expert Systems with Applications*, vol. 42, pp. 1166–1177, 2015.

[6] H. Syahputra and A. Wibowo, "Comparison of Support Vector Machine (SVM) and Random Forest Algorithm for Detection of Negative Content on Websites," *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI)*, vol. 9, no. 1, pp. 165–173, 2023.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[8] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," in *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 2019, pp. 6105–6114.

[9] C. Wang, M. Zhang, F. Shi, P. Xue, and Y. Li, "A Hybrid Multimodal Data Fusion-Based Method for Identifying Gambling Websites," *Electronics*, vol. 11, no. 16, p. 2489, 2022.

[10] A. S. Maldini, W. S. J. Saputra, and D. A. Prasetya, "Multimodal Detection of Covert Online Gambling Advertisements Using Faster R-CNN and Tr-OCR," *bit-Tech*, vol. 8, no. 1, pp. 953–963, 2025.

[11] I. W. B. Sentana, I. K. G. D. Putra, and A. A. K. A. Cahyawan, "Dataset Construction for Multimodal Detection of Online Gambling Advertisement Infiltrations on Websites," in *Proceedings of the International Conference on Informatics and Computing (ICIC)*, Atlantis Press, 2024.

[12] N. AlDahoul, M. J. T. Tan, H. R. Kasireddy, and Y. Zaki, "Advancing Content Moderation: Evaluating Large Language Models for Detecting Sensitive Content Across Text, Images, and Videos," *arXiv preprint arXiv:2411.17123*, 2024.

[13] X. Li, X. Xiao, J. Li, C. Hu, J. Yao, and S. Li, "A CNN-based misleading video detection model," *Scientific Reports*, vol. 12, p. 6092, 2022.

[14] M. Auer and M. D. Griffiths, "Using artificial intelligence algorithms to predict self-reported problem gambling with account-based player data in an online casino setting," *Journal of Gambling Studies*, vol. 39, pp. 1273–1294, 2023.