Integrating RAG with Face Recognition for Personalized Guest Services for Hospitality Industries

Munir Bin Rudy Herman, Pai Chet Ng, Malcolm Yoke Hean Low, Detlev Remy
Infocomm Technology Cluster, Singapore Institute of Technology, Singapore
2200963@sit.singaporetech.edu.sg, {paichet.ng, malcolm.low, detlev.remy}@singaporetech.edu.sg

***Abstract*—The hospitality industry has increasingly adopted advanced technologies to enhance guest experiences, yet many guests remain dissatisfied with on-site services despite effective pre-stay recommendation systems. Our proposed *AIReceiptionist* system integrates Face Recognition (FR) and Retrieval-Augmented Generation (RAG) modules to provide personalized and efficient guest services. The FR module identifies guests as they enter the public lobby, and the RAG module retrieves detailed guest information from the relevant database, generating prompts to notify staff. We present the system implementation of our *AIReceiptionist* by leveraging existing deep learning tools, including MTCNN for face detection and InceptionResnetV1 for face recognition, combined with OpenAIEmbeddings for data embedding and GPT-4o for language model responses. Using synthetic data tailored to hospitality operations, we validated *AIReceiptionist*'s performance and compared it with the existing GPT-4o model. The chatbot responses show that our *AIReceiptionist* enhanced by FR-RAG module significantly outperformed the generic GPT-4o in delivering personalized guest services, demonstrating its feasibility in enhancing guest satisfaction in hospitality environments.**

***Index Terms*—LLM, GPT-4o, RAG, personalized services, hospitality industry**

I. Introduction

The hospitality industry has significantly benefited from recommendation systems that assist guests in finding suitable accommodations based on their preferences. These systems leverage extensive data from social media platforms and user reviews to enhance the booking process by recommending the most appropriate options for travelers [1], [2]. Despite these advancements, there remains a notable gap between pre-stay recommendations and on-site services, as evidenced by the ongoing dissatisfaction among many guests with their vacation stay experiences [3]. This dissatisfaction highlights the need for continuous improvement in the services provided during the guest's stay.

Current technologies implemented in hospitality industries, such as the Internet of Things (IoT), Artificial Intelligence (AI), and robotics, have been employed to enhance guest experiences. These technologies aim to streamline operations and provide personalized services [4], [5]. For instance, smart hospitality technologies have significantly improved convenience and control for travelers, contributing to economic development and guest satisfaction [5]. However, there are still limitations with the current technologies implemented in the hospitality industries. Many stackholders rely on chatbots that are rudimentary and fail to assist staff effectively, lacking the capability to recognize loyal customers and provide timely information. Technologies like cameras or Bluetooth can identify guests as they enter the public space, but staff are often not immediately informed, and manual searches through guest profiles are required, causing delays and diminishing the seamless check-in experience [6].

Our proposed approach addresses these challenges by integrating Face Recognition (FR) with Retrieval-Augmented Generation (RAG) to empower an AI chatbot, namely *AIReceiptionist*. Unlike traditional chatbots that require staff to prompt them, our system actively responds as soon as a guest is recognized upon entering the public lobby. This integration enables the chatbot to instantly retrieve and provide relevant guest information, such as check-in details, booking history, and personal preferences, allowing staff to prepare and serve guests more efficiently. This proactive system enhances the overall guest experience by minimizing wait times and ensuring a more personalized service. Our proposed *AIReceiptionist* has the following contributions:

- **Integration of FR-RAG**: Combines FR technology with RAG to provide immediate guest information retrieval and notification to related staff.
- **Proactive AI Chatbot**: Develops a chatbot that actively provides assistance to related staff without requiring manual prompts, suited for a fast-paced hospitality environment.
- **Enhanced Guest Experience**: Improves the check-in process by reducing wait times and offering personalized interactions based on real-time data retrieval.

II. System Architecture

Fig. 1 shows our proposed system, *AIReceiptionist*, which includes a FR module to identify guests and a RAG module to retrieve detailed guest information. *AIReceiptionist* enhances the check-in process and overall guest experience by integrating advanced technologies for seamless and personalized service. The FR module detects and recognizes the guest's face, identifying them by name or indicating a new guest. Once identified, the RAG module retrieves relevant information from the relevant database, performs semantic search, and generates prompts to notify the related staff, ensuring they are well-informed and prepared to provide personalized service.

TENCON 2024 - 2024 IEEE Region 10 Conference (TENCON) | 979-8-3503-5082-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/TENCON61640.2024.10902965

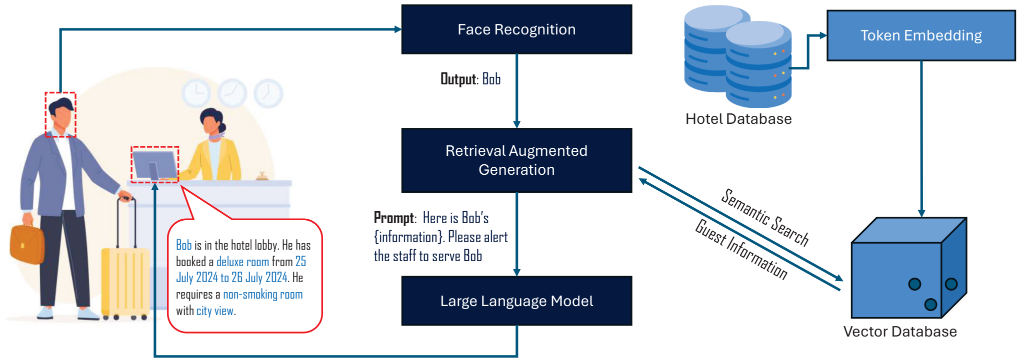

Fig. 1: Our proposed *AIReceiptionist* consists of two modules: a FR module to identify the guest who is entering the public lobby, and a RAG module to retrieve the information about the identified guest.

A. Face Recognition (FR) Module

Consider an input image $x \in \mathbb{R}^{w \times h \times c}$, with width w, height h and number of channels c, captured by the permise's entrance camera. We can describe the face recognition module based on the following 3 components:

- **Face Detection:** The input image x is processed to locate face regions using a face detection algorithm. Let $f_d(x)$ denote the face detection function, which outputs a bounding box $B = [p_w, p_h, l_w, l_h]$ around the detected face, i.e., $B = f_d(x)$, where p_w and p_h are the starting point of the bounding box, and l_w and l_h are the length in width and height.
- **Facial Feature Extraction:** The detected face region within the bounding box B is extracted and normalized to obtain x_B. A feature extraction model $f_e(x_B)$, parameterized by weight W, maps the normalized face image to a feature vector $v \in \mathbb{R}^d$, where d is the dimensionality of the feature space:
$$v = f_e(x_B; W)$$
The feature vector v is then compared against a database of known guest features using a similarity measure $sim(\cdot)$to identify the guest:
$$y = \arg\max_i sim(v, v_i)$$
where v_i are the feature vectors of known guests and y is the guest identity indicator, which can be the guest id stored in the related stakeholder's database.
- **Unseen Face:** If the similarity score for all comparisons is below a predefined threshold, the system flags the guest as new. In such cases, the guest is registered in the system, and their data is added to the recognition set for future identification.

B. Retrieval-Augmented Generation (RAG) Module

Once the guest is identified with y as the guest identity indicator (e.g., the guest ID stored in the relevant database), the goal is to perform a semantic search on the vector database to retrieve the guest information.

- **Vector Database Construction:** The vector database is constructed by embedding the available data in the relevant database. This process involves token embedding, where each piece of guest information is transformed into a high-dimensional vector. Let $info_i$ represent the information of guest i. An embedding function $embed(info_i)$ maps this information to a vector $u_i \in \mathbb{R}^d$. This process is repeated for all guest records, resulting in a vector database where each entry is a vector representation of the corresponding guest information.
- **Semantic Search and Information Retrieval:** To retrieve guest information, a semantic search is performed on the vector database using the identified guest's vector y. The query vector q is generated by embedding the guest's identity y to $q = embed(y)$. A semantic search is then conducted in the vector database to find the vectors most similar to q. This search retrieves the relevant guest information, such as check-in details, booking history, preferences, reviews, and dietary preferences.
- **Augmenting the Prompt for the LLM:** The retrieved information is crafted into a prompt input for the LLM. For example, if the identified guest is Bob, the prompt might be:"*Here is Bob's {information}. Please alert the staff to serve Bob.*" The LLM processes this prompt to generate an active alert for the related staff, ensuring they are well-informed and prepared to provide personalized service to the guest upon their arrival.

III. System Implementation and Comparison

A. Implementation

We leverage existing deep learning tools and models provided by OpenAI to implement the *AIReceiptionist* system. For the FR module, we use the Multi-task Cascaded Convolutional Networks (MTCNN) model [7] to detect faces in the input

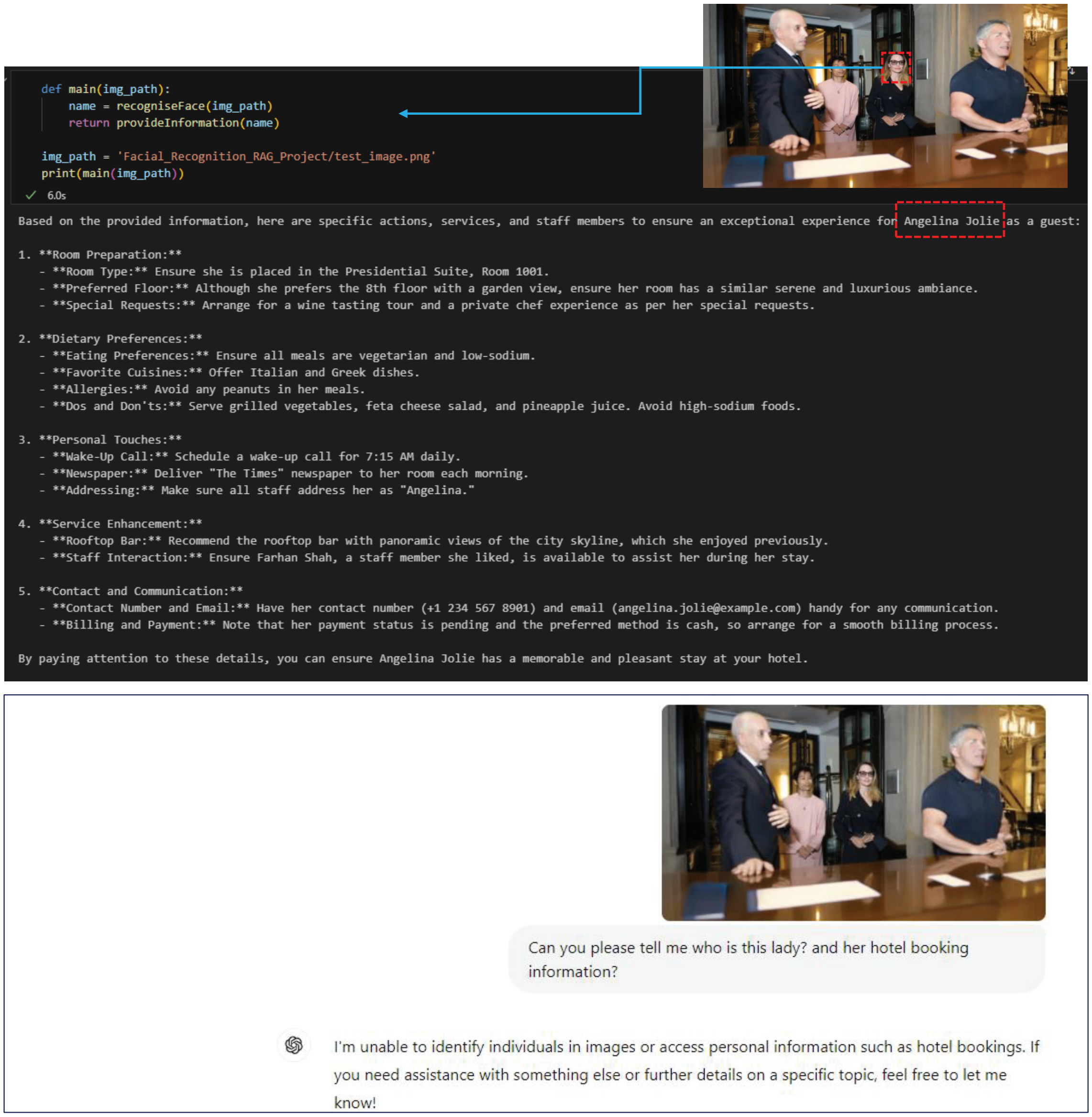

Fig. 2: The upper diagram shows the response of our *AIReceiptionist* when a guest arrived; and the bottom diagram shows the response of ChatGPT when we query the chatbot about the guest. Note that both platform used the same LLM, i.e., GPT-4o.

image, initialized with specific parameters for image size, margin, minimum face size, thresholds, and post-processing. The MTCNN model processes the RGB image, returning the cropped face image and detection probability. We then employ the InceptionResnetV1 model [8], pre-trained on the VGGFace2 dataset [9], for face recognition. This fine-tuned model generates a face embedding, which is compared against known embeddings using Euclidean distance to find the closest match. Known embeddings and corresponding names are pre-loaded, and the system identifies the guest with the closest match, calculating the confidence level based on the distance.

For the RAG module, data from multiple CSV files (booking information, customer preferences, feedback, financial records, guest information, housekeeping schedules, inventory, loyalty programs, room information, and service information) is loaded and combined into a single dataset. This data is embedded into high-dimensional vectors using the OpenAIEmbeddings model [10]. A vector store is created with DocArrayInMemorySearch [11] for efficient retrieval and configured as a retriever. Query processing and response generation involve structuring queries with a prompt template for the language model. Specifically, the GPT-4o model [12] is utilized to generate responses. When a guest is recognized, relevant information is retrieved through semantic search on the vector database, generating a prompt for the language model.

B. Synthetic Dataset

To verify the implementation of our *AIReceiptionist* system, we utilized synthetic data specifically tailored to meet our requirements. Existing data synthesis tools like the Synthetic Data Vault [13] use machine learning algorithms to create synthetic data from real data patterns. However, we opted to create our own synthetic data because these tools did not provide some critical information needed by our system and generally generated generic tabular data about guest booking information, which did not fully align with our needs.

To ensure our synthetic data met the specific requirements of *AIReceiptionist*, we generated data across multiple categories relevant to hospitality operations, including guest information, room information, booking and reservations, services and amenities, feedback and reviews, inventory and supply orders, loyalty programs, financial records, housekeeping, and customer preferences. The synthetic data was stored in .csv files to ensure comprehensive coverage of all necessary operational aspects. This approach allowed us to simulate and test various scenarios involving guest interactions and hospitality operations, leading to a robust evaluation of *AIReceiptionist*'s performance. By generating detailed and specific synthetic data tailored to our system's needs, we effectively addressed any gaps left by generic data synthesis tools, ensuring a thorough validation process for our system.

C. Performance Comparison

We compared our implementation using the underlying GPT-4o model with the existing GPT-4o model available on the ChatGPT platform. The existing GPT-4o model, when queried without the FR-RAG enhancement, provides generic responses that cannot identify individuals or access personal information from hospitality related databases, as shown in Fig. 2. This limitation prevents ChatGPT from delivering personalized service recommendations based on guest-specific data, as it lacks the ability to recognize individuals in images or retrieve specific personal details.

In contrast, the GPT-4o model enhanced with the FR-RAG module delivers highly personalized guest services. By integrating face recognition technology, the enhanced model identifies guests upon entry, ensuring their preferences and requirements are promptly addressed. Our *AIReceiptionist* recommended specific actions for Angelina Jolie, such as preparing her preferred room type, accommodating dietary preferences, and arranging personal touches like a wake-up call and newspaper delivery, as shown in Fig. 2. This enhanced performance allows hospitality staff to be proactively informed about guests' preferences and special requests, facilitating a more personalized and attentive service. This capability is particularly beneficial in hospitality industries where personalized guest experience is crucial.

IV. Conclusion

Our *AIReceiptionist* demonstrates its capability in providing immediate identification of guests and retrieval of relevant information, enhancing guest experiences with seamless check-in processes. Future work should incorporate real-time feedback to refine guest experiences and expand to multimodal data inputs like voice and gesture recognition. Enhancing scalability and integration with other hospitality management systems will ensure broader applicability and ease of adoption.

Acknowledgment

This research is supported by the Ministry of Education, Singapore, under SIT Ignition Grant (Grant number: IG (S) 3/2023 – 949).

References

[1] Jui-Hung Chang, Chen-En Tsai, and Jung-Hsien Chiang, "Using heterogeneous social media as auxiliary information to improve hotel recommendation performance," *IEEE Access*, vol. 6, pp. 42647–42660, 2018.

[2] Pai Chet Ng, James She, Ming Cheung, and Alexander Cebulla, "An images-textual hybrid recommender system for vacation rental," in *2016 IEEE Second International Conference on Multimedia Big Data (BigMM)*, April 2016, pp. 60–63.

[3] Mingyang Li, Yumei Ma, and Pingping Cao, "Revealing customer satisfaction with hotels through multi-site online reviews: A method based on the evidence theory," *IEEE Access*, vol. 8, pp. 225226–225239, 2020.

[4] Prasanna Kansakar, Arslan Munir, and Neda Shabani, "Technology in the hospitality industry: Prospects and challenges," *IEEE Consumer Electronics Magazine*, vol. 8, no. 3, pp. 60–65, May 2019.

[5] Gurucharan Singh, Sandeep Raheja, and Rakesh Sharma, "Elevating hospitality with smart hotel technologies: A guest – centric perspective," in *2023 IEEE Engineering Informatics*, Nov 2023, pp. 1–9.

[6] Sadrakh Richardson, Felicia Jovanka, Paulina Kurnia Zabrina, Maria Pia Adiati, and Dendy Rosman, "The consequences of digital concierge chatbots acceptance on hotel guest experience and satisfaction at ihg hotel and resort," in *2023 International Conference on Digital Applications, Transformation Economy (ICDATE)*, July 2023, pp. 1–5.

[7] Ning Zhang, Junmin Luo, and Wuqi Gao, "Research on face detection technology based on mtcnn," in *2020 International Conference on Computer Network, Electronic and Automation (ICCNEA)*. IEEE, 2020, pp. 154–158.

[8] Shuai Peng, Hongbo Huang, Weijun Chen, Liang Zhang, and Weiwei Fang, "More trainable inception-resnet for face recognition," *Neurocomputing*, vol. 411, pp. 9–19, 2020.

[9] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman, "Vggface2: A dataset for recognising faces across pose and age," in *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*. IEEE, 2018, pp. 67–74.

[10] OpenAI, "OpenAI Embeddings," https://platform.openai.com/docs/guides/embeddings, 2024, Accessed: 2024-07-14.

[11] LangChain Contributors, "Docarrayinmemorysearch," https://python.langchain.com/v0.2/docs/integrations/vectorstores/docarray_in_memory/, 2024, Accessed: 2024-07-14.

[12] OpenAI, "Hello GPT-4o," https://openai.com/index/hello-gpt-4o/, 2024, Accessed: 2024-07-14.

[13] Neha Patki, Roy Wedge, and Kalyan Veeramachaneni, "The synthetic data vault," in *IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, Oct 2016, pp. 399–410.