Discover Artificial Intelligence

Research

Automatic summarization of cooking videos using transfer learning and transformer-based models

P. M. Alen Sadique[1] · **R. V. Aswiga**[1]

Received: 30 July 2024 / Accepted: 16 January 2025
Published online: 25 January 2025

Abstract
The proliferation of cooking videos on the internet these days necessitates the conversion of these lengthy video contents into concise text recipes. Many online platforms now have a large number of cooking videos, in which, there is a challenge for viewers to extract comprehensive recipes from lengthy visual content. Effective summary is necessary in order to translate the abundance of culinary knowledge found in videos into text recipes that are easy to read and follow. This will make the cooking process easier for individuals who are searching for precise step by step cooking instructions. Such a system satisfies the needs of a broad spectrum of learners while also improving accessibility and user simplicity. As there is a growing need for easy-to-follow recipes made from cooking videos, researchers are looking on the process of automated summarization using advanced techniques. One such approach is presented in our work, which combines simple image-based models, audio processing, and GPT-based models to create a system that makes it easier to turn long culinary videos into in-depth recipe texts. A systematic workflow is adopted in order to achieve the objective. Initially, Focus is given for frame summary generation which employs a combination of two convolutional neural networks and a GPT-based model. A pre-trained CNN model called Inception-V3 is fine-tuned with food image dataset for dish recognition and another custom-made CNN is built with ingredient images for ingredient recognition. Then a GPT based model is used to combine the results produced by the two CNN models which will give us the frame summary in the desired format. Subsequently, Audio summary generation is tackled by performing Speech-to-text functionality in python. A GPT-based model is then used to generate a summary of the resulting textual representation of audio in our desired format. Finally, to refine the summaries obtained from visual and auditory content, Another GPT-based model is used which combines the output of the frame summary and audio summary modules and give the final enhanced summary. By minimizing the complications involved with traditional and sophisticated methodologies, this research helps with the development of a straightforward but efficient cooking video summarization system. The results achieved in the work are on par with the existing work in the respective field which demonstrates comparable performance and efficacy in converting cooking videos into detailed recipe texts.

Keywords Automated summarization · Transfer learning · Computer vision · Natural language processing · Speech recognition · Convolutional neural network · InceptionV3 · ChatGPT · Food recognition

✉ R. V. Aswiga, aswiga.rv@vit.ac.in | [1]School of Computer Science and Engineering, Vellore Institute of Technology, Chennai 600127, Tamil Nadu, India.

1 Introduction

Food preparation videos are now abundant on the internet, giving cooking video viewers around the world a fully immersive culinary experience. There are many different recipes for people to learn, catering to a diverse global population. The advent of the internet as a resource for culinary exploration has transformed people's perspectives and understanding of cooking, from master chefs sharing amazing recipes to amateurs sharing their simple recipes that are concise and instructive. The need for a fresh approach stands out due to immense complicated data and detailed and complex methods used in the current methods of transforming visually impressive food preparation videos into concise, learning focused written summaries.

The process of how these fascinating visual demonstrations of cooking could be converted into short and informative text recipes is challenging. This challenge is provoked by the very nature of the cooking videos themselves which is a kinetic blend of visual and auditory information. Even though models such as Transformer and LSTM (Long short-term memory) have shown success in NLP (Natural language processing) tasks, they present some computational overhead and require a considerable amount of training data readily available in different cooking styles and cuisines. It is anticipated that both LSTM and Transformer models will have difficulty identifying the specific points of variance or unpredictability presented by the food preparation situations. A wide range of ingredients, preparation methods, and presentation styles are commonly used in video recipes. These models can be challenging due to their complexity and diversity, which makes generalization difficult. Moreover, the large-scale characteristics and computational requirements of these models may pose difficulties, especially when real-time or resource-constrained processing is essential.

In machine learning, the concept of transfer learning refers to the process of using or adapting a model that has been trained on a particular task to perform other tasks that are related. The fundamental principle involves using what was learned during the training of a model on source domain and applying it towards improving learning or performance on target domain. In our work, transfer learning is employed by fine-tuning the Inception v3 model which is a pre-trained architecture which is trained on large dataset called ImageNet for the task of dish recognition. This involves optimizing the model's weights and updating or adding more layers to the existing architecture of model based on the food image dataset. Transfer learning has been utilized by summarization GPT models as well. These GPT models are pretrained on a wide range of language data to capture general linguistic patterns. These transformer-based models which are pre-existing are utilized in our project to produce the most precise and comprehensive texts by expanding their capability of natural language processing. We rely heavily on transformer-based models, which are renowned for their expertise in working with sequence data which help us create full descriptions of the content in cooking videos.

This work proposes a transfer learning and transformer-based system that takes cooking videos and turn them into easy-to-follow summary texts. Initially, Two CNN models and one GPT-based model is used for video frame summary generation. A pre-trained CNN architecture called Inception v3 is fine-tuned with food image dataset for the task of dish recognition and a custom-made CNN is built on fruits and vegetable image dataset for ingredient recognition. A GPT-based model is used here to combine the results obtained from both the CNN models to get a clearer summary from video frames. Afterwards, for audio summary generation, the audio will get converted from speech to text using python libraries. Then the converted text is given to a gpt module for generating a summary based on the information in text. Ultimately, Refinement of the summaries from the visual and auditory cues of the video is done by a gpt model by combining the outputs of the other two gpt models used in the former. All the gpt models' output are made to give in a uniform format so that the analysis and comparisons of results is effective.

The strength of our proposed method is its thoughtful fusion of sophisticated language models, audio processing methods, and image-based models. This process of harmonization guarantees that the intricacies included in cooking videos are effectively tackled, resulting in the production of recipe texts that are clear, condensed, and rich in context. The major contribution of the proposed work is to build a system that consists of:

i) Frame Summary Generation Module consisting of:

- One fine-tuned CNN (Inception v3) for dish recognition,
- One custom-built CNN for ingredient recognition, and
- GPT-based model-1 for combining the results of the two CNN models.

ii) Audio Summary Generation Module consisting of:

- Speech-to-text conversion of audio from the video, and
- GPT-based model-2 for summarizing the converted text content.

iii) Refinement of Summaries Module consisting of:

- GPT-based model-3 for combining the results of the above two modules to get a refined and enhanced summary.

The proposed work outperforms existing methods in video summarization, demonstrates the critical role of integrating audio and visual data for accurate results, and recognizes the ingredients effectively.

2 Related works

In recent years, the generation of automatic summaries from videos has been a point of focus in research, where multiple notable contributions were given which addressed the intricacies and the complexities of this task. Sobue et al. [1] introduced a semi-automatic system for generating the summary of cooking videos that make use of step-by-step recipe pictures to guide the process of making summaries, enhancing user interaction with cooking content. This method focuses on fulfilling the need for more engaging and visually pleasing cooking video summaries intended for social networking platforms where users want quick summaries of recipe tutorials that are easy to view and understand at a glance. The method of collecting feature vectors from culinary video frames and recipe photographs using an optimized Inception V3 model is explained in the paper. The chosen model is well-known for its remarkable accuracy in recognising objects in tasks, which makes it a suitable match for the application domain of identifying kitchen scenes. In order to alter how close or far apart distinct recipes were from one another, the triplet loss function was utilized to quantify the distance between feature vectors and modify these values based on patterns of cooking processes in photographs. Zhou et al. [2] took the advancements of transformer-based models a step further. They put forward a comprehensive video captioning model, built using a Masked Transformer. This model was not only good, but actually excellent, at suggesting event descriptions and their captions. Through this, they highlighted the potential of transformer-based systems in understanding video tasks. Similarly, Nishimura et al. [3] made their contribution in generating recipes from unsegmented cooking videos. They made it possible using a transformer-based joint approach which was used for event selection and caption generation. That's how it brought about a development in the ways to generate recipes from videos. Transformer models, like what proposed by Zhou et al. [2], may have great results, but they also present a challenge. They are hard to train, due to their vast number of parameters and intricate self-attention mechanisms. These intricate and complex elements can affect their efficiency, particularly in real-time tasks or settings with limited resources. Furthermore, transformers need plenty of labelled data for training which can pose a hindrance, especially in areas with a scarce amount of labelled data. Wu et al. [4] proposes a framework to improve the recipe generated from cooking videos by adding attention to ingredients. This feature is added in order to increase the quality of output captions by adding exact and fine-grained details.

The framework takes care of the issues with general video captioning method like overlooking sequential steps and shared ingredients. For instance, it utilizes context from previous steps to generate caption for the current step. Extensive experiments not only on YouCookII but Cooking-COIN dataset successively confirm the accomplishment of the approach by improved recipe quality via more effective ingredient recognition and contextual information integration.

Progress in multimodal methods have been one of the great developments in recipe generation based on various sources such as images of foods. Imam Saheb et al. [5] proposed a reverse cooking system that creates recipes from food photos along with enhancements in nutritional information estimation, recipe generation, and user preference. Through this, a fusion of multimodal data, that is textual and visual information has emerged as a major characteristic of contemporary techniques in recipe generation. Chhikara et al. [6] proposed FIRE, a multimodal method for generating recipes from food images, exhibiting versatility in the domain of food computing as well. An implementation of the proposed image-to-recipe generation using multi-model Architecture algorithm is discussed by Mahal et al. [7] which yielded impressive human-level results. These studies mainly underscore the importance of integrating diverse data modalities in achieving a more complete recipe comprehension from visual cues. Cross modal retrieval is one of the areas where research has been concentrated along with a focus on the encoding of cooking recipes in a structured format. Papadopoulos et al. [8] proposed "cooking programs" that was a representation of both the cooking recipe and food image, resulting in better cross-modal retrieval outcomes. This representation had structure because it contained sets of functions and their sequences, which could be understood as steps taken during cooking, creating a new way to comprehend recipes. They have also used another approach, developing an end-to-end model for cross-modal recipe

retrieval with the help of Transformer encoders. Demonstrating their cutting-edge achievements with respect to the Recipe1M dataset, this work considers Transformers as reliable means of learning textual recipes and visual food images.

Developments in detection of cooking state have resulted from the introduction and examination of new models in this domain. In this context, the work by Khan et al. [9] introduced a novel way for cooking state recognition, which utilizes Vision Transformers that achieved an astonishingly high level of accuracy on the Cooking State Recognition Challenge Dataset. It seems that the potential of Vision Transformers lies in recognizing various states and actions within cooking videos. On the other hand, Chen et al. [10] studied deep reinforcement learning designed for difficult sequential decision problems. They shed light on the problems and the possible ways that could be considered in the future when applying deep reinforcement learning to cooking video understanding. Furthermore, Doman et al.'s [11] Transformer model demonstrates how this type of model outperforms others by capturing long-term relations between elements and helping to reduce the model's complexity, which also brings a new perspective on video understanding architecture. It describes the concepts of multihead attention and scaled dot-product attention, which are essential components of the self-attention process.

Xu et al. [12] presents a novel Multimodal Attention LSTM (MA-LSTM) framework that uses temporal attention to generate captions for videos while seamlessly integrating multimodal data, such as frames, motion, and audio. Good performance was achieved over the popular benchmarks MSVD and MSR-VTT datasets because of the fusion unit and expanded attention mechanisms that also enable combining and exploration across multiple modalities. That being said, it is anticipated that the inclusion of numerous modalities and attention mechanisms would result in higher computing demands throughout the training and implementation phases of the model. It also might be challenging to get sufficient labeled data for training, especially for a domain with limited data availability. Özer et al. [13] proposed a video-captioning model using an architecture consisting of CNN and LSTM networks. The principal aim using this architecture was to provide video descriptions, satisfying the desires of visually impaired people. The presented architecture has delivered very encouraging results, but it requires substantial computational requirements. Furthermore, in real-world situations where video is of low quality, or in situations where other modalities, such as audio, convey more additional information, using visual cues alone as in the model is likely to be insufficient for an efficient captioning model.

Research on food recognition methods reveals the effectiveness of transfer learning in improving recognition accuracy. Yu et al. [14] proposes a CNN-based food recognition approach to boost accuracy in dietary assessments. It uses the Food-101 dataset to apply transfer learning to pretrained Inception V3 models and Inception ResNet. The method used in this work obtains an overall accuracy of 72.55% and a top-5 accuracy of 91.31%, proving that transfer learning greatly improves recognition ability. Possible future work includes tweaking network architecture to achieve a perfect score and deploying the algorithm on mobile platforms. A food recognition system based on smartphones is presented by Fakhrou et al. [15] for youngsters with visual impairments. It applies a deep CNN model trained on a custom dataset. Inception model is employed in the system and generates an accuracy of 95.55%. In order to facilitate transfer learning, pre-trained models that have been trained on the ImageNet dataset are used. Providing real-time food recognition during their dining boosts the confidence of visually impaired children from voice feedback. Since transfer learning works well in recognizing food items, using pretrained models and a big dataset like Food-101 makes it easier to train the model and helps in recognizing many categories of food within the videos.

Tang et al. [16] carried out one of the most comprehensive evaluations of the most recent advancements and challenges in large language model (LLM)-based video interpretation. In addition to providing a detailed explanation of the methods used in Video-LLMs, they also discussed the main tasks, datasets, and application domains that come under this category. Another important study, presented by Sahoo et al. [17], examined prompt engineering strategies to be used in conjunction with Vision-Language Models (VLMs) and Large Language Models (LLMs). First, they discussed several application areas and then looked at how these strategies have evolved from straight forward zero shot prompting to advanced techniques like Chain-of-Thought (CoT). By exposing the unrealized promise and problems in prompt engineering that still need to be fixed, this study aims to encourage more research. Paredes et al. [18] gives a full rundown on the Chat GPT API and how it is integrated into coding work for software development. It discusses how AI influences the global landscape, highlighting ChatGPT as a significant achievement in 2023. It is praised as it gives fast and right answers to all sort of questions. The paper emphasizes how valuable it is to bring the ChatGPT API into software creation to make the work better and efficient. These studies give knowledge about the integration of language models with our task of video summarization.

In reviewing the existing literature, the urgent need for an effective and simple summarization techniques was recognized to distill lengthy visual content into concise and actionable recipe texts. Research has identified important drawbacks in the literature, including the under-utilization of audio data of the video and the computational complexity

of existing models like transformer and LSTM models. These models have predominantly focused on visual cues alone, rather than considering a multi-modal approach. A methodical approach is proposed to address these issues, making use of knowledge from these studies to fill in the gaps that were found. Using a multi-modal approach, we included audio along with video frames to improve the summarizing process's comprehensiveness. By utilizing speech-to-text capabilities, we were able to add audio cues to the summary output, which improved the richness of the final recipe text. In order to reduce processing complexity associated with existing models due to the large amount of data and intensive processing capabilities required, we additionally simplified model architectures in our methodology. To achieve comparable performance, we combined more straightforward and simpler image-based models with GPT-based architecture along with audio processing. A workable way to turn cooking films into comprehensive recipe texts is provided through the proposed work, advancing the field and improving user experience and accessibility when processing visual culinary information.

This is how the remaining content of the paper is organized. Section 3 presents details of the proposed methodology. The experimental results and discussions are presented in Sections 4 and 5, respectively.

3 Materials and methods

The proposed method consists of three systematic steps; Video Frame Summary Generation, Audio Summary Generation and finally Fine-tuning the summaries generated from video frames and audio. Figure 1 illustrates the architectural framework of the proposed work, each component of which will be thoroughly discussed in the subsequent sections.

3.1 Video frame summary generation

This is the part where the information is extracted from the visual content which is obtainedfromthevideoframes.Theobjectivehereistorecognizeboththeingredients and the dish that has been prepared in the video. As most of the cooking recipe videos on the internet, it will begin with the ingredients required like the fruits or vegetables. The video will end displaying or tasting the dish prepared. So, we have two separate models dedicated for dish recognition and ingredient recognition respectively. These CNN models will recognize them from the frames. These models are trained with separate

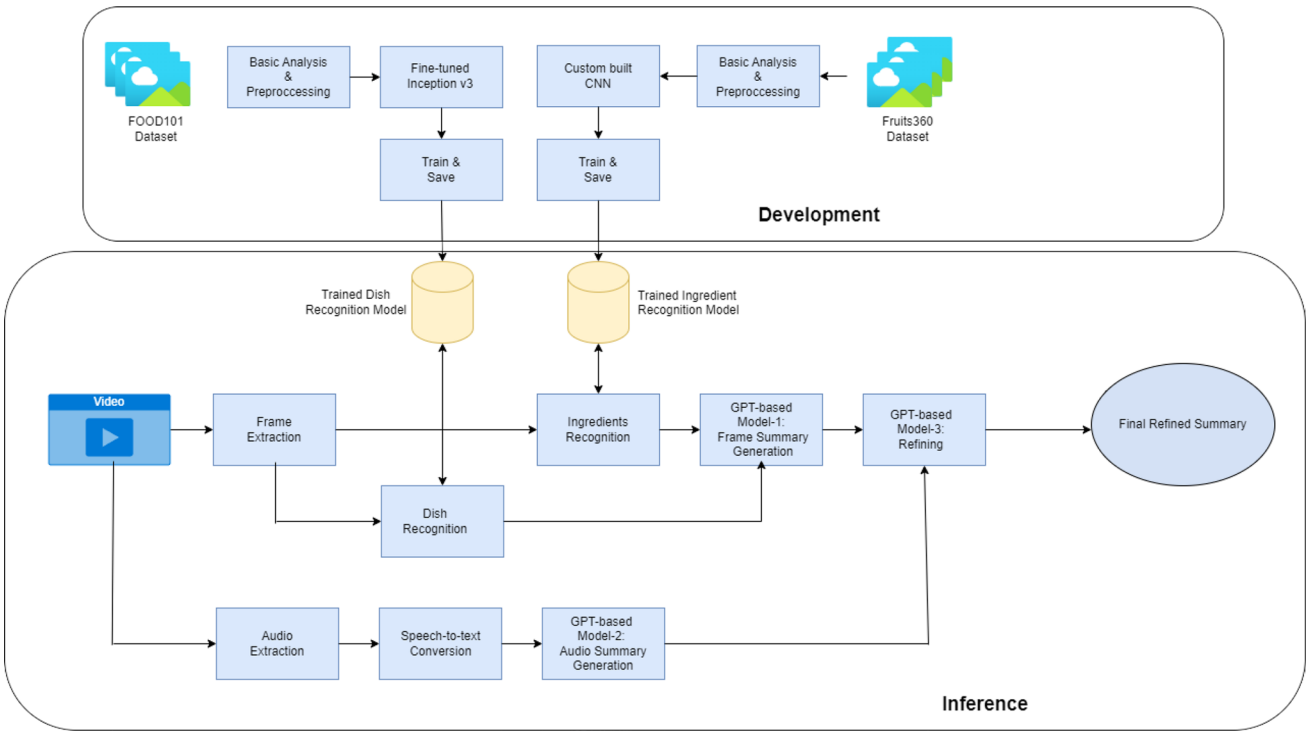

Fig. 1 Architectural diagram illustrating the proposed framework

datasets of food images and ingredients images which are images fruits and vegetables. The dish recognition CNN will be applied only at the final 20% of the video as the dish will be present at that part only. The Ingredients CNN will be applied to the first 40% of the frames as the ingredients will be present at that portion only. The final part of the frame information extraction is a customized GPT-based Module where we input the extracted dish and ingredients details and it will generate the detailed recipe based on the input. Tables 1 and 2 gives the details of the dataset used for frame summary generation process.

3.1.1 CNN for dish recognition

3.1.1.1 Dataset The popular FOOD101 dataset is used to train the CNN model for dish recognition.Thedatasetisfamousindeeplearningtasksthatinvolvefoodimages.Thereare 101 food categories and 101,000 pictures in the dataset. For each class, 250 manually reviewed test images are provided as well as 750 training images. The total size of datasetis5GB. This dataset is pre-processed and a basic data analysis is conducted to understand the data and is applied for the training. We used data augmentation which is using multiple copies/versions of data images during training to increase the training variability. The second dataset utilized in our study is the Fruits 360 Image dataset, which comprises 81,104 instances across 70 classes. This dataset has been divided into training and validation sets in an 85:15 ratio respectively.

3.1.1.2 Fine-tuned Inceptionv3 model A framework based on the InceptionV3 model which is already pre-trained is built for the task of dish recognition in frame summary generation. It is a model that has already been trained which is previously trained with a very big dataset called as ImageNet for understanding general visual features and as a result, makes an excellent starting point for image classification tasks due to its capacity in extracting hierarchical representation from images. To fine-tune this architecture specifically for dish recognition, we used Food-101 dataset that has 101 food classes. The approach consisted of importing relevant layers from Tensor Flow library and configuring Inception V3 model accordingly. To initiate the model's training with pre-trained weights drawn from the "imagenet" parameter that would give it a strong basis in terms of general image features, we had to use ImageNet's 'weights' parameter that is set to "imagenet". The top classification layer was excluded and defined an input shape which is compatible with our dataset. In the next step, more layers are added to the basic model like average pooling, dropout for regularization and fully connected dense layer with soft-max activation to predict food category probabilities. This augmentation has resulted in a model that can learn about minute details of different foods types hence making it a great enabler for accurate dish identification within food images. Consequently, we achieved successful transfer learning through fine-tuning from source domain (ImageNet) to target domain (Food-101) where it specializes in dish recognition while taking advantage of strong features learned during pre-training. The architecture of this process is illustrated in Fig. 2.

3.1.2 CNN for ingredient recognition

3.1.2.1 Dataset The Fruits360 images dataset, which consists of pictures of fruits and vegetables, is used to train the CNN model for ingredient recognition. The dataset comprises of different pictures of fruits and vegetables, amounting to 131 classes in total out of which only 70 classes are considered in the model building. All the images in this dataset have been normalized to a size of 100×100 pixels. During training, we used 60,486 images while an additional 20,618 were kept aside for testing. This extensive dataset is thus a valuable resource for training and evaluating models designed for fruit and vegetable recognition tasks.

3.1.2.2 Custom-made CNN model The Ingredient Recognition Model is built using a custom Convolutional Neural Network (CNN) architecture. CNN model is configured in such a way that it can effectively recognize different fruits and vegetables in images, with input shape normalized as (100, 100, 3) and can be classified into 131 classes present within the dataset. The model has several layers that start with three Convolutional 2D layers having 32, 32 and 64 filters respectively which is followed by ReLU activation function and Max-Pooling 2D for feature extraction and dimensionality reduction. Afterwards, the Flatten layer is used to change the two-dimensional feature maps to one-dimensional feature vector which is then connected to a Dense layer of 1024 units with ReLU activation. Using ReLU activation, it feeds the output from the preceding layer (which should have been flattened) via a fully connected layer having 1024 units. Dropout regularization is also used at this stage to prevent overfitting by randomly dropping out nodes at each training step including input nodes which are dropped at a drop-out rate of 0.5. Lastly, for classification, a Dense layer with an activation function named softmax is used, with the number of output units

Table 1 Literature review highlighting the strengths and weaknesses of different approaches

Approach	Key features	Strengths	Limitations
Wu et al. [4]	Attention mechanism focusing on ingredients, sequential steps, and shared ingredients for captions	Enhanced ingredient recognition and contextual integration for better captions	Primarily focused on caption generation; may not extensively address visual recognition
Imam Saheb et al. [5]	Multimodal fusion of food images and text; enhances nutritional information along side recipe generation	Produces comprehensive recipes by combining different data types	Quality depends on both textual and visual data; requires considerable data for fusion
Chhikara et al. [6]	Utilizes multimodal methods for generating recipes from food images	Versatile generation and potentially integrates various modalities effectively	Performance might vary based on data quality and modality integration
Papadopoulos et al. [8]	Cross-modal retrieval through structured representations of recipes and images using Transformer encoders	Improved comprehension and retrieval of cooking sequences	Complexity in model training; data integrity across modalities can be a challenge
Xu et al. [12]	Multimodal Attention LSTM framework integrating video frames, motion, and audio for captioning	Combines data for richer content generation and context	Increased computational cost and reliance on sufficient labeled data
Above exiting works incorporates Attention Mechanism and Multimodal fusion techniques			
Approach	Key features	Strengths	Limitations
Khan et al. [9]	Utilizes Vision Transformers for recognizing cooking states in videos	High accuracy in recognizing various cooking actions	High computational requirements; may not suit static recipe generation
Chen et al. [10]	Explores deep reinforcement learning for sequential decision-making in cooking video understanding	Addresses challenges in sequential contexts effectively	Complicated implementation and high resource requirements may hinder practicality
Ozer et al. [13]	Architecture using CNN and LSTM for making video descriptions accessible to visually impaired individuals	Specifically designed to cater to visually impaired users	Primarily uses visual cues, potentially neglecting other informative modalities
Yu et al. [14]	CNN-based food recognition using transfer learning on the Food-101 dataset	High accuracy in food recognition increases application viability	Performance may vary with dataset diversity; possible complexity in mobile deployment
Fakhrou et al. [15]	Mobile-friendly food recognition system utilizing a deep CNN model and transfer learning	Deep learning capabilities provide real-time food recognition	Performance may vary with diverse training datasets; might compromise model complexity for mobile use
Tang et al. [16]	Evaluates advancements in large language models (LLMs) for video interpretation	Promotes integration of language understanding with video content comprehension	High computational and data requirements; challenges in specific domains like cooking
Sahoo et al. [17]	Investigates prompt engineering with Vision-Language Models (VLMs) and LLMs for various applications	Advances in application strategies and understanding of user prompts	Yet to be fully addressed; challenges in prompt engineering still exist
Above exiting works incorporates traditional CNN and LSTM models for processing descriptions of cooking videos			

Table 2 Summary of datasets used for video frame summary generation

Dataset	Dataset name	Number of instances	Number of classes
1	Food101 Dataset	101,000	101
2	Fruits360 Image Dataset	81,104	70

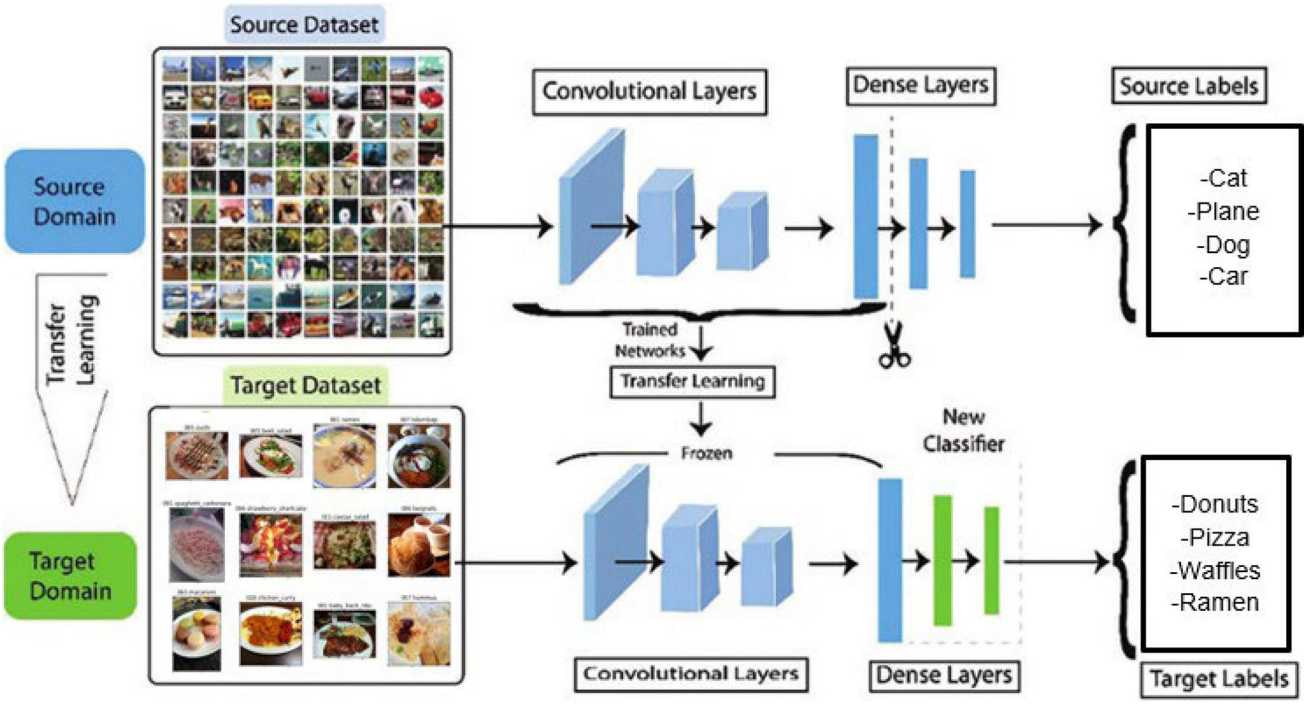

Fig. 2 Architecture of Dish Recognition Model

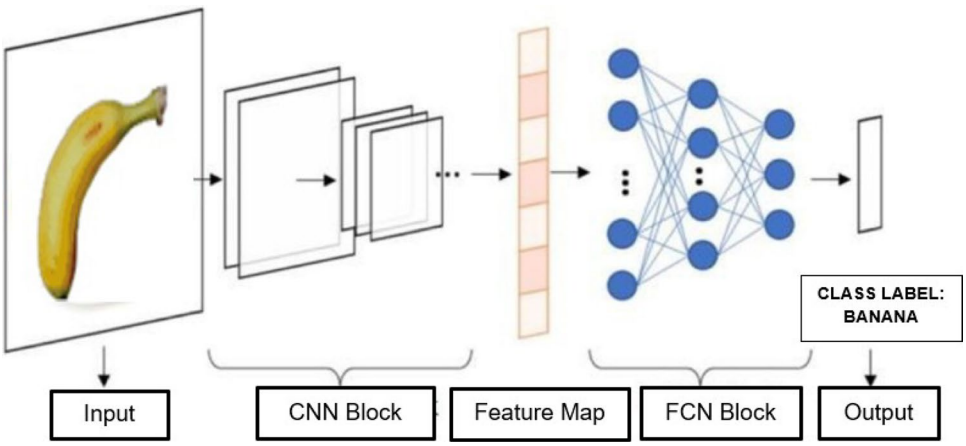

Fig. 3 Architecture of Ingredient Recognition Model

equal to the number of classes taken into consideration during the training phase. For optimization purposes while training the model, categorical cross-entropy loss and RMSprop optimizer are used. During each iteration of training process, batch size remains constant at 32 images per batch with parameters adjusted iteratively for hundred epochs so as to optimize performance. This CNN architecture demonstrates robust fruits and vegetable classification capabilities for the task of ingredient recognition. Figure 3 shows the architecture diagram of the custom-built CNN for ingredient recognition.

3.1.3 GPT-based model-1 for combining the results of dish recognition and ingredient recognition models

Both the Dish Recognition and Ingredient Recognition models are trained with sufficient data. Now, these trained models will be applied on the video frames extracted from videos to obtain visual data from the video. These visual data will be then given to the GPT-Based model-1 for summarizing the data and producing the recipe based on the data. ChatGPT, being a powerful language model capable of processing and summarizing textual input makes the task of visual data extraction from videos easier and effective. By making use of its natural language processing capabilities, the GPT-Based model gives rich summary of the data in the desired format. This GPT-Based model is accessed through its Application Programming Interface (API) using a unique API key which can be generated by any user. Architecture of this process is shown in Fig. 4.

3.2 Audio summary generation

The audio summary generation module consists of two steps; Firstly, Speech-To-Text conversion of the extracted audio from the video. Then, this converted text is given to a GPT-Based model for summarizing the audio content.

3.2.1 Speech-to-text conversion of audio

A core part of our workflow is the audio parsing section, which extracts important auditory information from video contents. We use Python to extract the audio part of videos with specific libraries and tools. By using Python Text To Speech (PTTS) functionality, we convert the audio content from video to textual content. It changes sound signals that are embedded in a video into textual form for easier processing and evaluation purposes. The textual version of the transformed audio becomes very vital; as inputs for other stages of our pipeline in recipe generation. In order to ensure accurate interpretation and capturing spoken instructions, descriptions and other auditory cues found in a video, we make use of PTTS functionality. The first step here is to use the Speech Recognition library in Python to convert the audio content into text. This is done by reading the audio file using Audio File class with in Speech Recognition library which later uses recognize google method to convert these to text. The next thing we do is to break up large audio files into smaller chunks using PyDub library based on the length of silence. This helps in efficient processing as it focuses on smaller pieces at a time. Each chunk then goes through analysis for speech contents individually with a function that splits the sound according to pre-defined parameters like minimum silence length and silence threshold. After processing, transcribing and combining texts of audio chunks, we finally have transcription of all spoken contents in the audio.

3.2.2 GPT-based model-2 for generating summary based on the converted text from speech

Once the audio is processed and text converted from speech is obtained from it, the text is then passed to GPT-Based Model-2 for audio summary generation that generates the summary based on the audio of the video. The result will consists of ingredients, summary in few sentences and step-by-step recipe. The result is obtained in such a way by making use of prompt engineering, which fine-tunes the output of audio summary generation module. The most frequent food item and ingredient recognized by the food recognition model and ingredient recognition model within the frames will only be passed to the GPT-based model to prepare summary. We instruct the language model to avoid or ignore any of the detected ingredients if they don't match with the food item recognized since there are chances for false recognition.

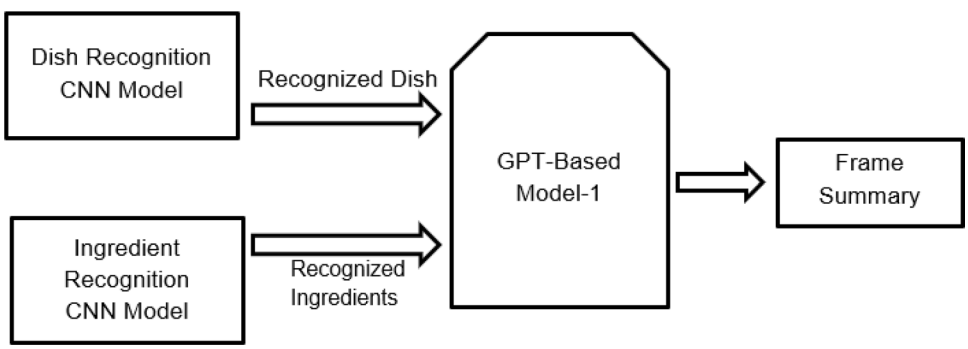

Fig. 4 Architecture of GPT-Based Model-1

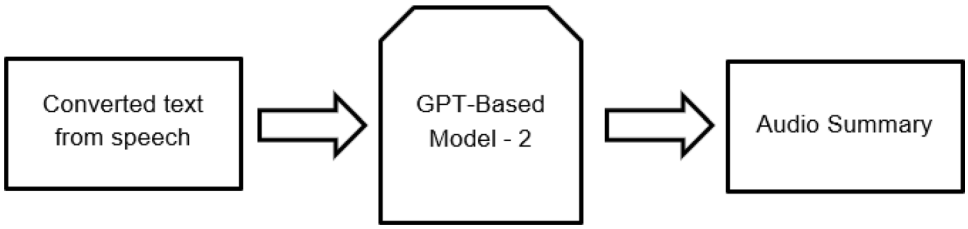

Fig. 5 Work flow of GPT-Based Model-2

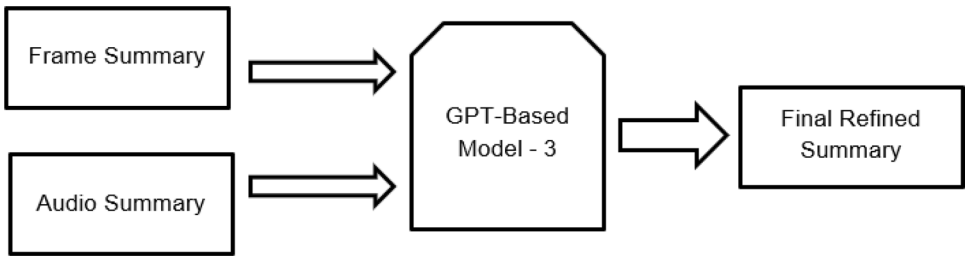

Fig. 6 Workflow of GPT-Based Model-3

Consequently, our system can extensively summarize detail audio recipes there by improving user experience and usefulness of our video analysis task. A work flow of this task is illustrated in Fig. 5.

3.3 GPT-based model-3 for refinement of the summaries obtained

Once the audio summary is generated from the converted speech-to-text data, it becomes the input of our Summary Refinement Module alongside video's frame summary that has been extracted from the video frames. The textual output of frame summary generation module and audio summary generation module is given to the GPT-Basedmodel-3 for a refined summary generation. The workflow of this module is illustrated in Fig. 6. This module is purpose fully designed to leverage on GPT API which comes in a version referred to as ChatGPT. This Summary Module relies on large language model techniques for a comprehensive processing of transcribed audio and frame information. As such, this module analyzes both visual and auditory prompts resulting in an elaborate summary that includes recipe instructions, ingredients and even cooking methods as desired. If the video has no audio speech content, then only the output of frame summary generation module will be considered as the final output as the system has to learn only from the visual cues of the video. The output produced by this Summary Module gives actionable insights that allow for easy access to relevant content. Our system blends the video's elements of sound and sight, so as to improve the overall user experience. This helps users to have a better understanding of what they were shown on how to prepare different dishes thus enabling them to cook for themselves and interact with video content more effectively.

4 Results

4.1 Evaluation metrics

The metrics chosen for evaluating the proposed dish and ingredient recognition models is Accuracy and the area under Receiver Operating Characteristic (ROC). Accuracy is the firstly evaluated for both dish recognition and ingredient recognition models. It is an indicator that gives an overall measure of the model's prediction. For dish recognition, accuracy is recorded among an additional accuracy metric that takes into consideration the frequency in which the correct label is among the model's five top predictions, termed as top-5 accuracy. Equation 1 gives the formula for accuracy. Moreover, ROC curve was used to evaluate dish recognition and ingredient recognition models. The area under ROC curve indicates the proportion of the trade-off between the false and true positive rates. Formula for TPR (true positive rate) and FPR (false positive rate) is given in Eqs. 2 and 3. The ROC curve is usually plotted for all the classes in case of binary and multi-class classifications with few classes. As the models we trained have more than 50 classes, we plot the graphs as a single curve by taking averages of TPR (true positive rate) and FPR (false positive rate) of all the classes in the dataset. This approach is inspired from the one vs all method. Plotting a single curve makes it easy and simple to read it. Furthermore, we examined loss metric called categorical cross-entropy (CCE) loss, to measure how well the model reduces the prediction error

during training. Formula for this is as shown in Eq. 4. Specifically, for both models, we monitor the loss and accuracy trends over epochs using training and validation datasets, visualizing them through loss and accuracy plots. In training, we have employed the RMS prop (Root Mean Square Propagation) optimizer for the ingredient recognition model and SGD (Stochastic Gradient Descent) optimizer to the dish recognition model. These optimizers update the model parameters during training to reduce the value of the loss function which in turn improves the prediction of the model.

$$\text{Accuracy} = \frac{TrueP + TrueN}{TrueP + TrueN + FalseP + FalseN} \tag{1}$$

$$\text{TPR} = \frac{TrueP}{TrueP + FalseN} \tag{2}$$

$$\text{FPR} = \frac{FalseP}{TrueN + FalseP} \tag{3}$$

- TrueP: Number of positive samples that were correctly predicted.
- TrueN: Number of negative samples that were correctly predicted.
- FalseP: Number of positive samples that were incorrectly predicted.
- FalseN: Number of negative samples that were incorrectly predicted.

$$\text{CCE Loss} = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij}\log(p_{ij}) \tag{4}$$

- N is the number of dataset samples.
- y_{ij} is a binary function taking the value of 1 if the sample i belongs to class j and 0 otherwise.
- p_{ij} is the probability that sample i belongs to class j.

For evaluating the generated refined text summary, we make use of multiple metrics. These metrics evaluates the generated text by comparing it to a ground truth provided after our thorough study of the input video. ROUGE (Recall-Oriented Understudy for Gisting Evaluation) is a commonly used measure that includes ROUGE-1, ROUGE-2, and ROUGE-L as well as other sub-metrics. ROUGE-1 assesses the degree to which the generated summary and the ground truth summary coincide in unigrams, while ROUGE-2 expands this analysis to bigrams. However, ROUGE-L provides a recall-oriented evaluation strategy by taking into account the longest common sub-sequence between the candidate and reference summaries. Equation for ROUGE calculation is given in Eq. 5. We also use METEOR (Metric for Evaluation of Translation with Explicit ORdering), an important statistic that evaluates the quality of generated text by considering recall and precision. To account for linguistic variances, METEOR uses synonym matching and stemming in addition to implementing a penalty factor for words that are not aligned properly. Formula for METEOR is given in Eq. 6. We also make use of another popular metric called BLEU (Bilingual Evaluation Understudy), which determines how accurate the generated summary is for n-grams when compared to the reference or ground truth. The precision scores for unigrams to 4-grams are represented by BLEU-1 to BLEU-4, respectively which provide a thorough evaluation of the text's fluency and sufficiency. Formula for BLEU is given in Eq. 7. These metrics are essential for assessing the effectiveness of the final refined summary in our work. The values obtained for these metrics will be given and discussed in the subsequent sections.

$$\text{ROUGE} = \frac{\text{count}_{\text{match}}(n\text{−grams})}{\text{count}_{\text{total}}(n\text{−grams})} \tag{5}$$

- $\text{count}_{\text{match}}(n\text{-grams})$ represents the count of matching n-grams between the candidate and reference summaries.
- $\text{count}_{\text{total}}(n\text{-grams})$ represents the total count of n-grams in the reference summaries.

$$\text{METEOR} = (1-\rho)\cdot\text{Precision}\cdot\text{Recall}\cdot\left(\frac{\text{Precision} + \beta\cdot\text{Recall}}{1+\beta^2}\right) \tag{6}$$

- Precision represents the precision score.
- Recall represents there call score.
- ρ represents the penalty for matches.
- β is a parameter for balancing precision and recall.

$$\text{BLEU} = \text{BP} \times \exp\left(\sum_{n=1}^{N} w_n \cdot \log p_n\right) \tag{7}$$

- BP represents the brevity penalty term.
- w_n represents the weight associated with n-grams.
- p_n represents the precision of n-grams.

4.2 Video frame summary generation

4.2.1 Dish recognition model

Promising results are obtained by the dish recognition model trained with inception-v3 architecture. It can be seen that validation loss of the model keeps falling over iterations and this means that the model is acquiring better accuracy and convergence towards optimal parameter settings. The preservation of model weights at this stage was possible due to epoch12's improvement from 0.9362 to 0.9230 for its validation loss. In subsequent epochs, there was a steady refinement of the model's performance evidenced by a decrease invalidation losses. For example, during epoch15, the validation loss fell to 0.8876 which represents an impressive improvement in model performance. This model performs well with its validation accuracies ranging between 77 and 78% thereby effectively distinguishing between items of food it is made to identify correctly. Furthermore, validation top-5 accuracy of 93.37% shows that this approach has ability to predict correct class within top five predictions with high confidence levels indicating robustness. Although some upsets are experienced in several epochs, overall trend demonstrates steady improvement and movement toward best results. The model has achieved commendable accuracy metrics, with the final accuracy reaching 78.01% on the validation dataset. The model achieved an AUC-ROC of 0.787, indicating its capability to discriminate between different dish categories. Figure 7a, b shows the accuracy curve and loss curve of the food recognition model respectively. The ROC Curve is given in Fig. 7c and the test results obtained are given in Fig. 7d.

4.2.2 Ingredient recognition model

Significantly, the ingredient recognition model achieved remarkable accuracy and validation loss metrics after 100 epochs of training. The model consistently demonstrated excellent performance despite fluctuations in subsequent epochs. For example, epoch 96 showed that the model has consistent accuracy amidst variable validation losses by having a high training accuracy of 96.94% and a validation accuracy of 94.00%. Finally, in terms of admirable efficacy measures, the last epoch recorded had an accuracy of 95.94% for training and a validation accuracy of 94.5%. We've also got area under ROC curve as 0.837. These findings highlight the robustness of this food recognition system and hence its suitability for practical applications including culinary analysis and identification of ingredients in food. Figure 8a, b shows the accuracy curve and loss curve of the ingredient recognition module. Figure 8c shows the ROC Curve and the test results of the model are shown in Fig. 8d. The Training and validation metrics of the two CNN models are given in Table 3.

4.2.3 GPT-based model-1 for combining the results of dish recognition and ingredient recognition models

Using the findings from the dish recognition model, which accurately identifies and categorizes food item shown in the video frames, and the ingredient identification model that captures principal ingredients, our frame summary generation process utilizes these insights to produce detailed informative summaries. Our system's combination of these models assists in extracting important information from video frames including identified dishes and their respective ingredients. Afterward, making use of a modified GPT (Generative Pre-trained Transformer) API specifically called chatGPT, our Summary Module synthesizes this information into comprehensive summaries. These summaries include crucial details such as recipe instructions, ingredients, and cooking methods, thus providing users with coherent insights on the cookery

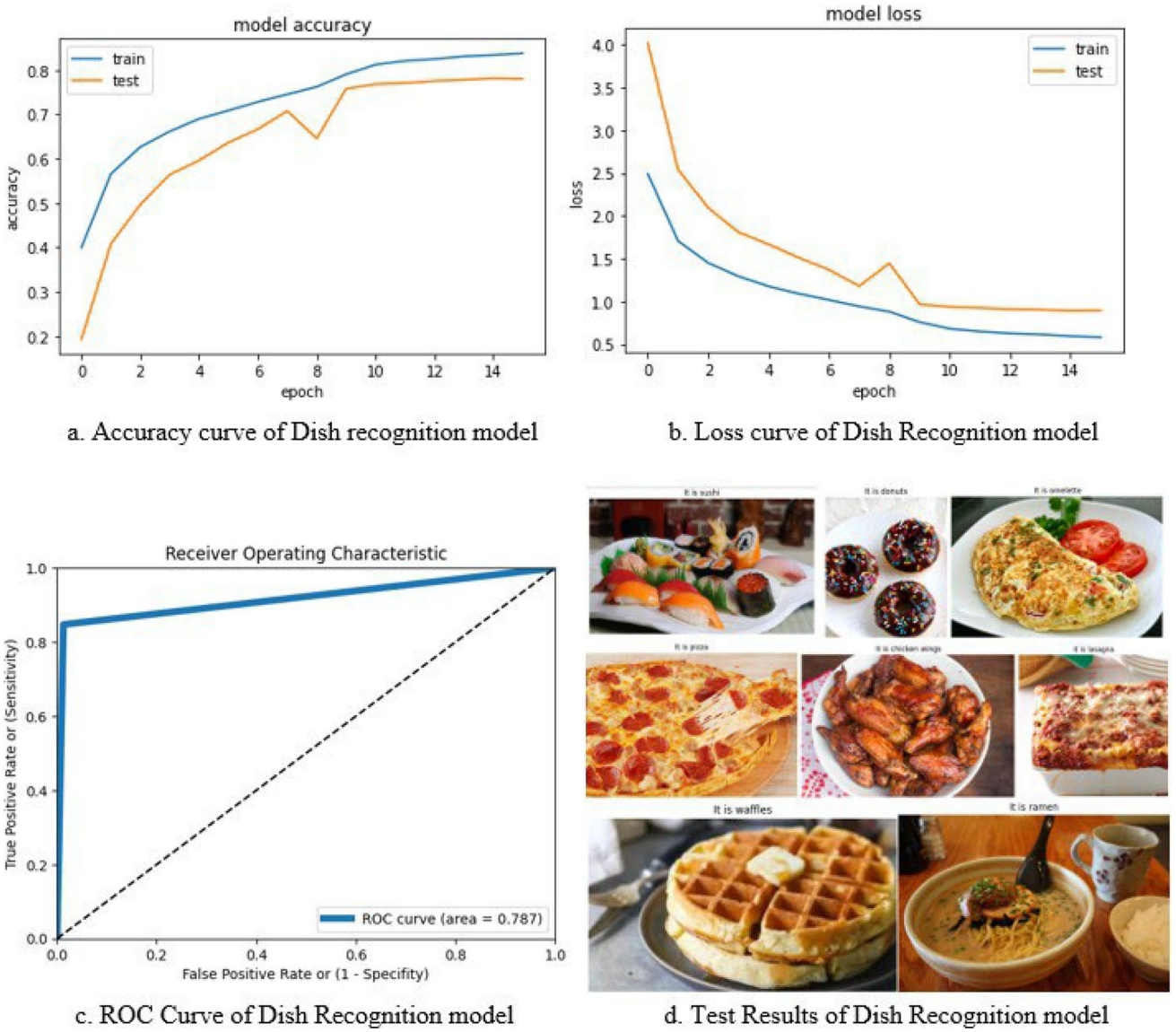

Fig. 7 **a** Accuracy curve of Dish Recognition model. **b** Loss curve of Dish Recognition model. **c** ROC curve of Dish Recognition model. **d** Test results of Dish Recognition model

show presented in a given video. The integration of food and ingredient recognition outcomes into frame summary generation process enhances the completeness as well as accuracy in summary development.

The result produced by the GPT-Based model-1 is illustrated in Table 4.

4.3 Audio summary generation

Using advanced speech recognition mechanisms, We produce word-for-word presentations of conversations and narrations in the video as transcriptions which may include cooking methods in relation to recipes as well as the ingredients' descriptions. We use this transcribed audio content in addition to the visual information obtained from video frames as a basis for summary generating process. To make our generated summaries more detailed and inclusive, we have incorporated sound processing techniques that give a rich understanding of what is happening during these video demonstrations. The transcribed content is passed to chatGPT and it's instructed to produce a detailed a summary based on the audio input. Table 5 illustrates the text parsed from the audio (transcriptions) and the summary generated by GPT-Based model-2 is shown in Table 6.

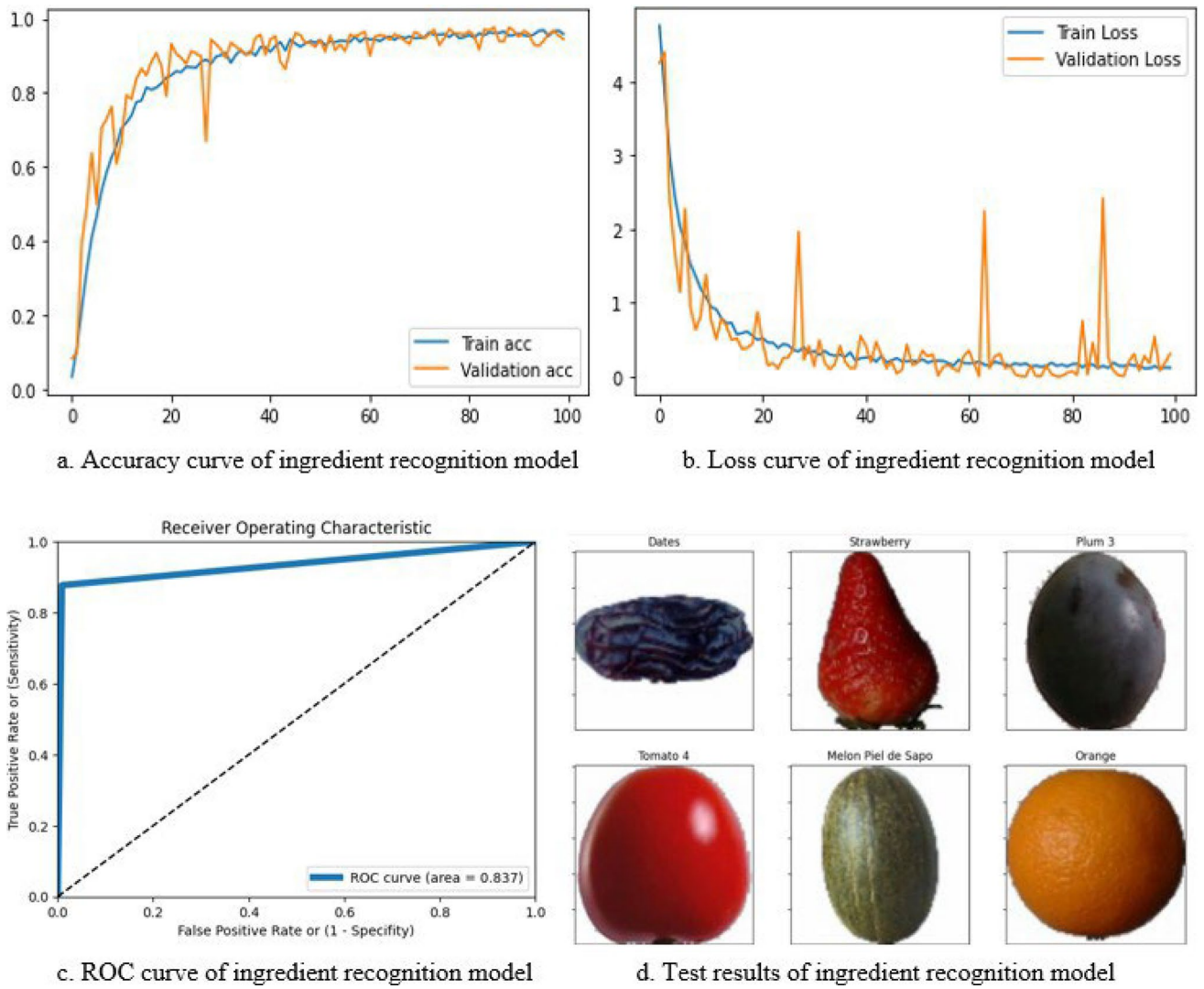

Fig. 8 **a** Accuracy curve of Ingredient Recognition model. **b** Loss curve of Ingredient Recognition model. **c** ROC curve of Ingredient Recognition model. **d** Test results of Ingredient Recognition model

Table 3 Performance of the two CNN models

Model	Epochs	Train Loss	Train Acc	Train Top-5Acc	Valid Loss	Valid Acc	Valid Top-5Acc	ROC curve area
Dish Recognition Model	16	0.5808	83.74%	96.30%	0.8917	78.01%	93.37%	0.787
Ingredient Recognition Model	100	0.1222	95.94%	–	0.3122	94.50%	–	0.837

4.4 GPT-based model-3 for refined summary generation

This module synthesized summary of information that encapsulates important details about cooking, including recipes, ingredients, cooking techniques and examples and other narrations from the videos through the use of a combination of video frames and the transcribed audio content. Advanced algorithms and machine learning methodologies are applied to harmonize visual cues with auditory cues in order to provide coherent but detailed summaries that allow users a more nuanced understanding of the culinary demonstrations in these videos. The summaries generated by using audio content and video frames are passed to this module to obtain a clearer and more concise version of the recipe generated from both the modalities. This fusion of modalities deepens the scope

Table 4 Summary generated by GPT-based model-1

Ingredients:
Tomato 2, Tomato Cherry Red

Summary:
Caprese salad is a simple Italian salad made with fresh tomatoes, mozzarella cheese, basil, Olive oil, and balsamic glaze. It is a light and refreshing dish perfect for summer

Stepbysteprecipe:
1. Slice the tomatoes and fresh mozzarella cheese into equal thickness
2. Arrange the tomato slices on a serving plate, alternating with the mozzarella cheese slices
3. Garnish the salad with fresh basil leaves
4. Drizzle olive oil and balsamic glaze over the salad
5. Season with salt and pepper to taste
6. Serve the caprese salad immediately as a side dish or appetizer. Enjoy!

Table 5 Transcriptions of audio chunks

Audio Chunk	Transcription
chunk1.wav	Cuz i want to show you how to make this real simple caprese salad and it starts with our grape tomatoes i'm just going to split it in half you don't have to leave them hold but i think by cutting i'm going to last that marinade to get inside and just really flavor them
chunk2.wav	I'm using these little mozzarella pearls you can use the boccaccini there just a little bit bigger or you can grab a whole fresh mozzarella ball and just cut it into a small dice next time i'm going to hit it with some fresh cracked pepper kosher salt my extra virgin olive oil and some balsamic vinegar that's perfect then one last final touch we're going to add some basil to that that's going to just give this a real nice pop so I have some fresh leaves i'm going to roll them up and slice them this is called a chiffon
chunk3.wav	That's going to be it forth is salad
chunk4.wav	Guys give that a little stir you can serve this over lettuce or eat it just like that and it's absolutely delicious and super refreshing thanks for watching cheers every one

Table 6 Summary generated by GPT-Based model-2

Ingredients:
Grape tomatoes, Mozzarella pearls or boccaccini, Fresh cracked pepper, Kosher salt, Extra virgin olive oil, Balsamic vinegar, Fresh basil leaves

Summary:
This recipe is for a simple caprese salad made with grape tomatoes, mozzarella pearls, fresh basil,
And a simple dressing of olive oil and balsamic vinegar

Step by step recipe:
1. Start by cutting the grape tomatoes in half
2. If using mozzarella pearls, keep them whole. If using boccaccini or fresh mozzarella ball, cut into small pieces
3. Place the tomatoes and mozzarella in a bowl
4. Season with fresh cracked pepper and kosher salt
5. Drizzle with extra virgin olive oil and balsamic vinegar
6. Roll up the fresh basil leaves and slice them thinly, then add to the salad
7. Gently toss the salad to combine all the ingredients
8. Serve over lettuce or enjoy as is for a refreshing and delicious caprese salad

of the summaries generated which gives users a holistic view on their contents. The summary generated by this model is evaluated using multiple metrics. The metrics used are ROUGE-1, ROUGE-2, ROUGE-L, METEOR, BLEU1, BLEU-2, BLEU-3 and BLEU-4. These metrics are calculated by taking the summary generated by GPT-based model-3 as candidate and the ground truth as reference. Input video frames are shown in Fig. 9. The values obtained for each of the mentioned metrics are given in Tables 7 and 8 gives the refined summary generated by GPT-Based Model-3 along with its ground truth crafted after the thorough study of the video.

Fig. 9 Input Video Frames

Table 7 Performance metrics of generated text compared to ground truth references

Metric	Score
ROUGE-1	0.8235
ROUGE-2	0.6723
ROUGE-L	0.8151
METEOR	0.8227
BLEU-1	0.8038
BLEU-2	0.7276
BLEU-3	0.6658
BLEU-4	0.6103

5 Discussion

As evidenced by the results obtained in our work, our approach has yielded promising performance in generating summaries for cooking videos. As mentioned in the previous sections, our approach combined computer vision techniques along with audio processing and pre-existing transformer based models to generate coherent summaries of cooking videos. The frame summary was generated successfully using a combination of the two CNN models and the GPT-Based model-1. The dish recognition model using fine-tuned inception v3 witnessed excellent performance, evidenced by the high validation accuracies and a steady reduction in validation loss over epochs. This image recognition model attained noteworthy performance, giving an accuracy of 78.01% on the validation set along with an ROC curve area of 0.787. Then, the second CNN model, the model for ingredient recognition has also performed really well with a validation accuracy of 94.5% along with an ROC Curve area of 83.7. These metrics implies that these models are able to classify food items in the images accurately which are the input to our GPT-Based model-1 for generating the frame summary which combines the output of both the CNNs.

Parallelly, our audio summary generation module has made significant strides. It starts with the efficient speech-to-text conversion of audio extracted from videos. Audio speech content is then transcribed to textual form using Python libraries and tools. The converted text is given to GPT-Based Model-2 for audio summary generation. This GPT-Based model successfully generated summary based on the speech in the audio which contains important information about the cooking process and thus we obtain a coherent summary of the video based solely on the audio content.

The refined summary generated by the GPT-Based model-3 is evaluated using multiple metrics for assessing the quality of the summary generated. This summary generated is compared with a ground truth summary which is developed by us after studying the test video thoroughly and then different metrics are calculated. The results obtained in Table 7 depict the promising performance of our approach with almost all metrics. Metrics like ROUGE, BLEU, along with its variants, and METEOR have continuously resulted in excellent scores, which reflect the effectiveness of our approach concerning the existing methodologies. The comparison of the scores of the proposed model with the previously tested scores from other papers related to video captioning tasks is given in Table 10. From the Table 10, it is visible that our proposed model has given better results than the existing works introduced in the domain of video captioning.

We also explored a scenario where audio information is unavailable where the frame summary generated itself serves as the final summary. To evaluate the effectiveness of this summary, we have taken the result obtained in

Table 8 Comparison of generated and reference summaries

Summary generated by GPT-Based Model-3 (Candidate)

Ingredients:
• Tomato 2
• Tomato Cherry Red
• Grape tomatoes
• Mozzarella pearls or boccaccini
• Fresh cracked pepper
• Kosher salt
• Extra virgin olive oil
• Balsamic vinegar
• Fresh basil leaves

Summary: Caprese salad is a classic Italian dish made with fresh tomatoes, mozzarella cheese, basil, olive oil, and balsamic glaze. This recipe combines the traditional caprese salad ingredients with grape tomatoes and mozzarella pearls or boccaccini for a twist on the original

Step by step recipe:
1. Slice the tomatoes and fresh mozzarella cheese (if not using pearls) into equal thickness. If using mozzarella pearls, keep them whole
2. Cut the grape tomatoes in half and place all the tomatoes and mozzarella in a bowl
3. Arrange the tomato slices on a serving plate, alternating with the mozzarella cheese slices
4. Season the tomatoes and mozzarella with fresh cracked pepper and kosher salt
5. Drizzle olive oil and balsamic vinegar over the salad. Use balsamic glaze if preferred
6. Roll up the fresh basil leaves and slice them thinly. Add the sliced basil to the salad
7. Gently toss the salad to combine all the ingredients
8. Serve the caprese salad immediately as a side dish or appetizer. Enjoy the refreshing and delicious flavors!

Ground Truth (Reference)

Ingredients:
• Grape tomatoes
• Mozzarella pearls or boccaccini
• Fresh cracked pepper
• Kosher salt
• Extra virgin olive oil
• Balsamic vinegar
• Fresh basil leaves

Summary: The Caprese salad, a classic Italian favorite, is made with fresh tomatoes, mozzarella cheese, basil, olive oil, and balsamic glaze. This recipe introduces grape tomatoes and mozzarella pearls for a unique twist on the original

Step by step recipe:
1. Cut the grape tomatoes in half and take the mozzarella pearls as whole. If not using mozzarella pearls, slice it into equal thickness
2. Place all the tomatoes and mozzarella in a bowl
3. Arrange the tomatoes on a serving plate with the mozzarella cheese pearls or slices
4. Season the tomatoes and mozzarella cheese with fresh cracked pepper and kosher salt
5. Drizzle extra virgin olive oil and balsamic vinegar over the salad
6. Roll up the fresh basil leaves and slice them thinly, then sprinkle over the salad
7. Gently toss the salad to combine all the ingredients and ensure the dressing is evenly distributed
8. Serve the caprese salad immediately as a side dish or appetizer. Enjoy the delicious and colorful combination of flavors!

Table 9 Comparison of evaluation metrics for video with audio (Frames + Audio) and without audio (Only Frame)

Metric	With audio	Without audio
ROUGE-1	0.8235	0.6010
ROUGE-2	0.6723	0.3173
ROUGE-L	0.8151	0.5803
METEOR	0.8227	0.3932
BLEU-1	0.8038	0.3367
BLEU-2	0.7276	0.2534
BLEU-3	0.6658	0.2043
BLEU-4	0.6103	0.1668

Discover Artificial Intelligence (2025) 5:7 | https://doi.org/10.1007/s44163-025-00230-y

Table 10 Comparison of proposed method with state-of-the-art methods

Method	Dataset	BLEU_1	BLEU_2	BLEU_3	BLEU_4	METEOR	ROUGE-1	ROUGE-2	ROUGE-L
Proposed method	Food101 and Fruits360	0.8038	0.7276	0.6658	0.6103	0.8227	0.8235	0.6723	0.8151
Encoder-Decoder Model [4]	YouCook2	–	–	0.145	0.092	0.170	–	–	0.379
	Cooking-COIN	–	–	0.456	0.445	0.295	–	–	0.529
LSTM-G [12]	MSVD Dataset	0.728	0.578	0.468	0.364	0.286	–	–	–
LSTM-C [12]		0.752	0.602	0.482	0.364	0.293	–	–	–
LSTM-V [12]		0.751	0.594	0.476	0.363	0.285	–	–	–
M-LSTM (G + C) [12]		0.801	0.681	0.586	0.492	0.326	–	–	–
MA-LSTM (G + C) (child-sum) [12]		0.823	0.711	0.618	0.523	0.336	–	–	–
Memory augmented recurrent transformer [3]	YouCook2	–	–	–	0.019	0.080	–	–	–
LSTM-Hybrid CNN [13]	MSVD Dataset	0.606	0.381	0.250	0.149	0.182	–	–	0.520
Universal Attention Transformer [19]	MSRVTT dataset	–	–	–	0.43	0.278	–	–	0.609
Vision transformer and reinforcement learning [20]	MSVD	–	–	–	0.565	0.364	–	–	0.728
	MSRVTT dataset	–	–	–	0.420	0.288	–	–	0.620
Multi-modal fusion LSTM [21]	Activity Net Captions dataset	0.167	0.082	0.403	0.191	0.102	–	–	–

Table 4 as the candidate and our ground truth as the reference for calculating metrics. The results obtained in this evaluation along with the results obtained when audio is available are given in Table 9 for comparison. These metrics shows how the performance of the system varies based on the varied input conditions. The findings suggest that the presence of multi-modal data significantly enhances the performance of our system. Conversely, when audio data is lacking, there is a discernible decrease in performance.

While we have been able to achieve quite remarkable successes in our work, there are certain limitations of our work that need to be acknowledged. One of the main limitations is there cognition scope of the ingredient, considering that, at the moment, we have only focused on fruits and vegetables. The model does not recognize anything other than fruit or vegetable. Adding to it, another problem here is that the dataset is comprised of images mostly of whole fruits and vegetables, thus the system is unable to detect processed foods such as diced, sliced, or chopped ingredients. Thus, although our ingredient recognition model is capable of classifying basic and whole food items, it is not meant for processed recipe ingredients. Similarly, Video clips without audio make the task of summarization less accurate, since the system uses both visual and audio cues for a refined summary. In the cases where audio is in existent, the summaries are based completely on visual interpretations and can thus be incomplete and in accurate which is evidenced by Table 9. Therefore, both audio and visual information is necessary for accurate and faithful summarization. These constraints point to the need for persistent research and development initiatives to make the system perform better and overcome the challenges encountered with cooking video analysis (Table 10).

6 Conclusion

Thus, our study offers a straightforward method for automatically summarizing cooking videos by fusing image-based CNN models, audio analysis, and GPT-based text refinement.Wedevelopedasystemthatiscapableofrecognisingdishandingredients in the video along with the auditory information available in it. The development of summaries enhanced with recipe details and cooking directions is made possible by the merging of visual and auditory information (multi-modal approach). Our results show promising outcomes, confirming the effectiveness of our method in extracting pertinent information from video frames with high validation accuracies and considerable area under ROC curve. Excellent results were also obtained by our method for the final refined summary obtained across metrics, such as ROGUE, BLEU, and METEOR. The notable innovation in our project lies in the usage of combination of visual and audio data along with the usage of pre-existing large language models based on GPT which makes the process of generation and refinement of summaries easier and effective. Future work will focus on expanding the scope of ingredient recognition to encompass processed forms and a wider variety of food items and expanding dish recognition to a wider range of cuisines which would enhance the system's applicability to a broader range of recipes. Additionally, enhancing the quality of audio integration offers a chance to improve the efficacy and accuracy of generated summaries especially in situations where audio is either nonexistent or of low quality and hence the summary primarily relies on the visual cues. Iterative refining procedures and user feedback systems can also improve the system's flexibility and responsiveness to user demands and preferences. The proposed work outperforms existing methods in video summarization, demonstrates the critical role of integrating audio and visual data for accurate results, and recognizes the ingredients effectively. For future work, multiple processed food dataset in diverse languages will be trained on Quantum models tailored for the diverse linguistic datasets.

Author contribution Manuscript was well written by Alen under the guidance of Dr. Aswiga R V. Both contributed equally for technical implementation and novel idea sharing.

Funding No funding was received to assist with the preparation of this manuscript.

Data availability No datasets were generated or analysed during the current study.

Declarations

Competing interests The authors declare no competing interests.

References

1. Sobue R, et al. Cooking video summarization guided by matching with step-by-step recipe photos. 2019, 16th International conference on machine vision applications (MVA). IEEE, 2019.
2. Zhou L, et al. End-to-end dense video captioning with masked transformer. Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.
3. Nishimura T, et al. Recipe generation from unsegmented cooking videos. arXiv preprint arXiv:2209.10134. 2022.
4. Wu J, et al. Ingredient-enriched recipe generation from cooking videos. Proceedings of the 2022 international conference on multimedia retrieval. 2022.
5. Saheb S, et al. Recipes creation using food images through inverse cooking. J Algebr Stat. 2022;13(3):2391–6.
6. Chhikara P, et al. FIRE: food image to recipe generation. Proceedings of the IEEE/CVF winter conference on applications of computer vision. 2024.
7. Mahal R, et al. Image-to-recipe translation using multi-model architecture. Image 7.05. 2020.
8. Papadopoulos DP, et al. Learning program representations for food images and cooking recipes. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022.
9. Khan AM, et al. Rethinking cooking state recognition with vision transformers. 2022, 25th International conference on computer and information technology (ICCIT). IEEE, 2022.
10. Chen J, Yin Y, Xu Y. Recipe snap—a light weight image-to-recipe model. arXiv preprint arXiv:2205.02141. 2022.
11. Doman K, et al. Video cooking: towards the synthesis of multimedia cooking recipes. Advances in multimedia modeling: 17th international multi-media modeling conference, MMM 2011, Taipei, Taiwan, January 5–7, 2011, Proceedings, Part II 17. Berlin, Heidelberg: Springer; 2011.
12. Xu J, et al. Learning multimodal attention LSTM networks for video captioning. Proceedings of the 25th ACM international conference on multimedia. 2017.
13. Özer EG, et al. Deep learning based, a new model for video captioning. Int J Adv Comput Sci Appl. 2020;11(3).
14. Yu Q, Mao D, Wang J. Deep learning based food recognition. Technical report, Stanford University; 2016.
15. Fakhrou A, Kunhoth J, Al Maadeed S. Smart phone-based food recognition system using multiple deep CNN models. Multim Tools Appl. 2021;80(21):33011–32.
16. Tang Y, et al. Video understanding with large language models: a survey. arXiv preprint arXiv:2312.17432. 2023.
17. Sahoo P, et al. A systematic survey of prompt engineering in large language models: techniques and applications. arXiv preprint arXiv:2402.07927. 2024.
18. Paredes CM, Gallardo CM, Claudio YMS. ChatGPT API: brief overview and integration in software development. Int J Eng Ins. 2023;1(1):25–9.
19. Im H, Choi Y-S. UAT: universal attention transformer for video captioning. Sensors. 2022;22(13):4817.
20. Zhao H, et al. Video captioning based on vision transformer and reinforcement learning. Peer J Comput Sci. 2022;8:e916.
21. Huang X, et al. Fusion of multi-modal features to enhance dense video caption. Sensors. 2023;23(12):5565.