Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 7.056

IJCSMC, Vol. 13, Issue. 9, September 2024, pg.9 – 18

Device for Spoken Word and Captioning of Illustrations

[1]Boriya Siddhant; [2]Sunil Kumar

[1,2]Department of Computer Science and Engineering
[1,2]Ramaiah University of Applied Sciences, Bangalore, India
Email: [1] siddhantbharvad@gmail.com; [2] skp.narayan@gmail.com

DOI: https://doi.org/10.47760/ijcsmc.2024.v13i09.002

Abstract:
This project shows the creation of a unique app that can take a picture as input, make a descriptive description for it, and then turn that caption into speech. The system uses cutting-edge models for both picture captioning and text-to-speech synthesis. This makes sure that the process from capturing an image to producing sound is smooth and quick. To do this, the app uses Hugging Face Transformers for model inference, which lets you use models that have already been trained and makes it easier to add advanced deep learning methods. The web interface was made with Streamlit, which makes it easy for people to connect with each other. A number of tools were also used to handle the processing of both audio and video data. This made sure that the application worked well at all times. In terms of accessible technology, this app is a big step forward. It gives people who are blind or have low vision a useful tool that can help them interact with visual material in a more meaningful way. This app is a great way to help bridge the gap between visual and auditory material because it uses cutting-edge technology and is designed to be accessible to everyone.

Keywords: CNN, Deep Learning, Text to Speech, Microsoft Speech TS, Speech Recognition.

1. Introduction

Computer vision has come a long way in the last few years, making it possible for computers to analyses and understand pictures in ways that were once thought to be impossible. Image captioning, which means writing a description of an image, is one of the most important areas of study in this field. This is useful in many situations, such as making content more accessible for people who are blind or visually impaired and better optimizing content with images for search engines. A lot of research and study has been done in the area of image labelling or getting information from machines that work like the mind. It is easy for people to explain pictures or scenes, but it's hard to get machines to do the same. The process of image captioning involves adding specific information and a description to a picture. The picture is made up of different types of elements, such as the objects, the background, and the relationships between the people or scenes shown. In the same way, language is a way to give important information about the scenes in the pictures and explain them. By making comments from the pictures, you can get information about them and get a quick idea of what life is like around the world. The picture description system may one day help people who are blind or have low vision "see" the world. It has become one of the most important topics in computer vision [6] and is getting more and more attention. As part of our suggested method, we include a way to get the image captions. The system uses the Flicker 8K dataset to get vector information of images and partial vectors of words, then combines them to make the perfect captions from the images. It also has a mechanism that turns the outputted caption into speech, which helps people who are blind or have low vision get information from the world. They mostly talked about three ways to explain images in this paper: CNN-RNN-based, CNNCNN-based, and reinforcement-based models. They gave examples of works, evaluation criteria, and the pros and cons of each of the three methods.

To make reading easier for blind people, Doshi et al. wrote a study in 2018 that made this suggestion. Using the Google Cloud Vision API, which can recognize text in a variety of situations, the system pulls text from images. The answer is in the form of a JSON structure. The g TTS engine is used to take the text and turn it into speech. After that, the text is saved as an mp3 file and played on an mp3 player. Finally, the text output is turned into synthetic speech audio output.

Vinyl et al. [3] suggested the Encoder-Decoder structure for jobs like adding captions to pictures. A big difference between them is that they use different types of convolutional neural networks (CNNs) to extract and store image data. The method has given researchers a way to start looking into the job of writing image

captions. This model uses a convolutional neural network to process information. It pulls information from images and sends it to a decoder that is made up of long short-term memory (LSTM). Lastly, the decoder takes the data apart and makes captions.

Baradar et al. [4] made a Flask app with a machine learning model in 2019. This model makes the subtitles by using the VGG16 and the Convolution neural network to pull out the image's features, which are then fed into the Recurrent Neural Network. And finally, these words are used to make the image's description. That's where the Flickr-8k file comes in handy. Based on a user's search query, this method gets the pictures.

In this work [5], Aker and Gaizauskas showed a way to automatically add captions to geo-tagged images by collecting information about image locations from several web pages. They used higher ROUGE scores than n-gram models, which led to dependency relational patterns. They had a dependency pattern model that was based on different cases.

In this summary, Horang [6] Wang and his team put together all the parts of the image caption generation task and talked briefly about the model framework that has been suggested in recent years as a way to solve problems. The framework is mostly about the algorithms of the different mechanisms, and the team gave a way for the system to use the attention mechanism. They made a list of the big datasets and the evaluation factors that are usually used in methods for making image captions.

According to Yang et al. [7], they came up with a method that can automatically make a natural image, which helps people understand the images. They came up with a multimodal neural network model that can find objects, do module localization in a way that is similar to how people do it, and get image descriptions based on that. In this study [8], they created a model using Long Short-Term Memory and a Convolutional Neural Network model. Then We used CNN with LSTM [9] to create an image caption machine. CNN pulls out the picture vectors, and LSTM pulls out the word vectors. Usually, image captioning and speech generation are done separately using different systems. Each task (image captioning and text-to-speech conversion) [10] is taken care of separately. Most of the time, these systems don't work together, so users have to directly send data between them. Also, many of the systems that are already in place might not use the newest developments in deep learning models [12], which could lead to less accurate captions and poorer speech synthesis.

2. Theory of Proposed Algorithm

In this paper we suggested system combines the tasks of creating speech from images into a single app. It uses Microsoft's SpeechT5 model and a HiFiGAN vocoder to turn text into speech [11] that is of high quality and Salesforce's BLIP model to make picture captions. Adding these models together makes it possible for spoken descriptions of pictures to be generated automatically, which improves accessibility and the user experience.

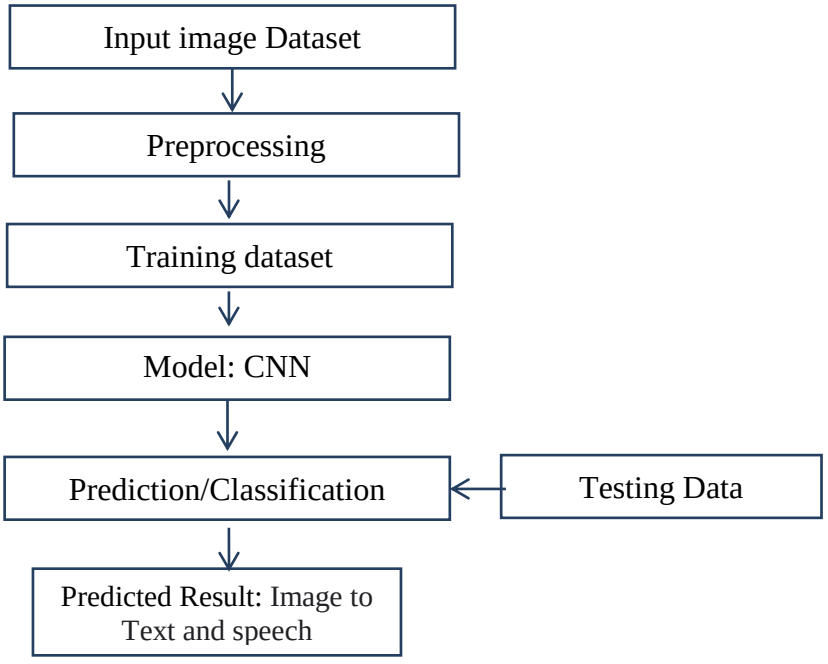

Figure 1. Proposed Architecture

3. Model Description

This paper shows how a system was built that combines image captioning and speech synthesis. The goal was to make a tool that is interactive and easy to use for making both audio and video material. The system is built around two main tasks: giving images written descriptions and turning those descriptions into speech that is synthesised. The execution starts with setting up the machine learning models that are needed. The system uses the BLIP processor and model for picture captioning. These are loaded and cached at the start to make caption generation go quickly. At the same time, the SpeechT5 processor, model, HiFiGAN vocoder, and speaker embeddings are initialised and cached to get ready for the voice synthesis part. These parts work together to make sure that the text that is created during the

picture captioning process can be turned into natural-sounding speech of high quality. There are two main ways for users to add pictures to the system: directly from a local device or through an external URL.

As soon as a picture is chosen, the system processes it and shows it in the user interface. The BLIP model then makes a text caption that correctly describes what the picture is about. This caption is shown to the person right away [13,14,15]. The speech synthesis module then works on the text that was created. When the SpeechT5 model and the HiFiGAN vocoder work together, they turn the text into speech, making a clear and natural sounding result. The system gives the user input by playing the synthesised speech and showing them the speech waveform, which gives them a better understanding of how the sound is put together.

The whole process is built into a Streamlit-based user interface that is meant to be easy to understand and use. The interface walks the user through each step—uploading a picture, making a caption, playing audio, and seeing the waveform—so the experience is smooth and interactive. This application shows how well image captioning and speech synthesis technologies can work together, which opens the door for more progress in multimodal content creation systems in the future.

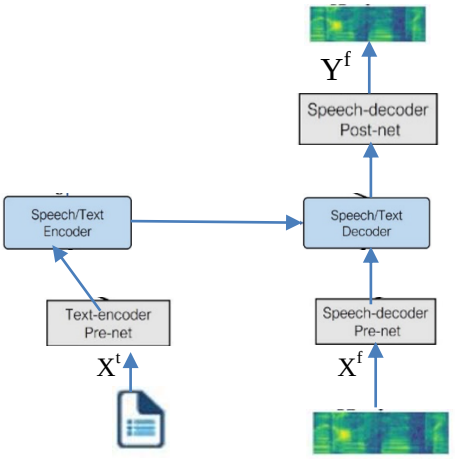

Figure 2. Text To Speech (TTS) architecture

We propose a unified-modal SpeechT5 framework, inspired by the success of T5 (Text-To-Text Transfer Transformer) in pre-trained natural language processing models, to investigate encoder-decoder pre-training for self-supervised speech/text representation learning. The SpeechT5 structure comprises of a common encoder-decoder organization and six modular explicit (discourse/text) pre/post-nets. In the

wake of preprocessing the information discourse/text through the pre-nets, the common encoder-decoder network models the arrangement to-succession change, and afterward the post-nets produce the result in the discourse/text methodology in light of the result of the decoder. Utilizing huge scope unlabeled discourse and message information, we pre-train SpeechT5 to gain proficiency with a brought together modular portrayal, wanting to further develop the displaying capacity for both discourse and message. We propose a cross-modal vector quantization method that randomly mixes up speech/text states and uses latent units as the interface between the encoder and decoder to align the speech and textual information into this unified semantic space. Broad assessments show the predominance of the proposed SpeechT5 system on a wide assortment of communicated in language handling undertakings, including programmed discourse acknowledgment, discourse combination, discourse interpretation, voice change, discourse improvement, and speaker ID.

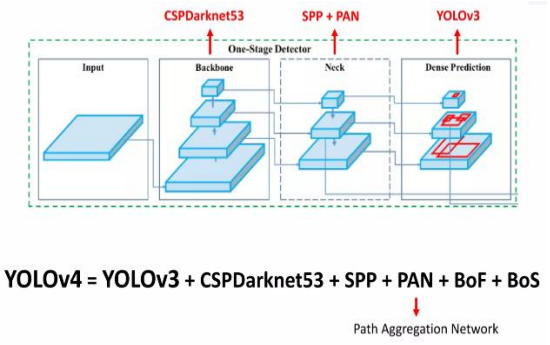

Figure 3. YOLOv4 Selection of architecture

The YOLO (You Only Look Once) algorithm represents a significant advancement in object detection within the field of computer vision [16]. Renowned for its exceptional speed and accuracy, YOLO has become a pivotal tool in various applications, including autonomous driving and system monitoring. This paper delves into the design, performance, and impact of the YOLO algorithm, illustrating how it continues to redefine the boundaries of real-time object detection.

This paper shows how to build a YOLOv4-based object recognition system that uses the OpenCV deep learning module to find things in real time [17]. The system is meant to find and place multiple items in an image quickly and accurately, which are qualities of the YOLO family of algorithms. When the system starts up, it loads the pre-trained YOLOv4 model, which includes its settings, weights, and class names. OpenCV's cv2.dnn.readNetFromDarknet function is used to load the

configuration and weights files, which describe the network's design and learnt parameters. The names of the model's layers are retrieved, and the network's unconnected output layers are used to find the output layers that make the final estimates. A file is read, and the class labels that show the groups that the model can find are put into a list. The process of finding objects starts with preprocessing the picture that was sent in. The cv2.dnn.blobFromImage function turns the picture into a blob, which is a format that the YOLO model can handle. The picture is made normal, shrunk to 416x416 pixels, and turned into a 4D blob by this function. The blob is then sent to the YOLOv4 model [18], which makes predictions across all of its output levels through a process called "forward pass."
The model gives information about each detected item, such as its bounding box coordinates, class scores, and confidence level. The most useful information is taken from these products through processing. For each item that is found, the algorithm figures out the centre coordinates, width, and height of the bounding box in relation to the original image dimensions. Then, the coordinates of the bounding box are changed to fit the size of the image. The method uses non-maximum suppression (NMS) [19] to improve the detections. NMS gets rid of unnecessary overlapping boxes and keeps only the most confident ones. This step is very important for lowering false hits and improving the accuracy of detection. The system then sends back a list of detections, with the object's class name, confidence score, and bounding box coordinates for each one. An empty list is returned if no things are found. This version shows how well YOLOv4 works for real-time object detection tasks, showing how it can balance speed and accuracy in real-world situations

4. Simulation and Results Analysis

Experiments were performed with using the state-of-the-art Hugging Face Transformers, the application has accurately referenced titles for numerous images and your home is guaranteed and moving. The program effectively implemented an application that takes an image as input, creates a presentation title, and converts the title to speech. The system demonstrated time efficiency, reducing the need for extensive computing resources while maintaining high accuracy. The user interface developed with Streamlit provided a simple and easy-to-use experience, allowing users to quickly upload images and receive associated explanations and explanations modular components designed the app, facilitating the modification of the underlying models for future development of machine learning and natural language applications In conclusion, the project provides a functional application

that serves as an interface communication services that are highly advanced, fulfill their objectives, and incorporate cutting-edge AI technologies. As a result, they gain experience.

Caption:
a man in a suit and tie sitting on the hood of a car

Audio:

0:00 / 0:03

See visualization

Average Detection Accuracy: 0.96

Caption:
a man riding a motorcycle down a street

Audio:

0:03 / 0:03

See visualization

Average Detection Accuracy: 0.89

Caption:
a man watering his garden with a watering can

Caption:
two young men sitting on a bench in the park talking and having a conversation

Audio:

0:03 / 0:03

See visualization

Average Detection Accuracy: 0.89

Fig 4. Showing Text-to-speech translation for the purpose of image identification and captioning as the final result

5. Conclusion

By successfully adding advanced deep learning models, we have shown that a seamless image-to-speech application is not only possible but also very useful. The system offers a unique way to make spoken descriptions straight from visual material by combining image captioning with text-to-speech generation. This new idea could make things easier for everyone, but it could be especially helpful for people who are blind or have trouble seeing because it would let them connect with visual information in a more meaningful way. This technology can be used for more than just accessibility; it can also improve user experiences in many other areas as well. For example, it can be used to support different learning styles in education by providing audio descriptions of visual materials. The project's success shows how powerful it can be to use cutting-edge machine learning methods together to make answers that solve problems in the real world. In the future, researchers could look into making these models even better by adding support for multiple languages or even making the system work for real-time uses. This would make the technology more useful to a wider range of people.

References

[1]. https://data-flair.training/blogs/deep-learning-project-ideas/
[2]. https://data-flair.training/blogs/python-based-project-image-caption-generator-cnn/
[3]. https://www.analyticsvidhya.com/blog/2020/11/create-your-own-image-caption-generator-using-keras/
[4]. https://machinelearningmastery.com/calculate-bleu-score-for-text-python/
[5]. Exploring Nearest Neighbor Approaches for Image Captionin
[6]. Image Captioning: Transforming Objects into Words
[7]. Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset for Automatic Image Captioning
[8]. Boosting Image Captioning with Attributes∗
[9]. Image Caption Generator Using Deep Learning
[10]. H. Fang, S. Gupta, F. Iandola, R. Srivastava, L. Deng, P. Dollar, J. Gao, X. He, M. Mitchell, J. Platt, et al. From captions to visual concepts and back. In CVPR, pages 1473– 1482, 2015.
[11]. Ao, J., Wang, R., Zhou, L., Wang, C., Ren, S., Wu, Y., Liu, S., Ko, T., Li, Q., Zhang, Y. and Wei, Z., 2021. Speecht5: Unified-modal encoder-decoder pre-training for spoken language processing. arXiv preprint arXiv:2110.07205.
[12]. Y. Gong, Y. Jia, T. Leung, A. Toshev, and S. Ioffe. Deep convolutional ranking for multilabel image annotation ICLR, 2014.
[13]. Y. Gong, L. Wang, M. Hodosh, J. Hockenmaier, and S. Lazebnik. Improving image sentence embeddings using large weakly annotated photo collections. In ECCV, pages 529– 545. Springer, 2014.
[14]. K. Gregor, I. Danihelka, A. Graves, and D. Wierstra. Draw: A recurrent neural network for image generation. arXiv preprint arXiv:1502.04623, 2015.

[15]. A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In CVPR, June 2015.

[16]. Jiang, P., Ergu, D., Liu, F., Cai, Y. and Ma, B., 2022. A Review of Yolo algorithm developments. *Procedia computer science*, *199*, pp.1066-1073.

[17]. Yu, J. and Zhang, W., 2021. Face mask wearing detection algorithm based on improved YOLO-v4. *Sensors*, *21*(9), p.3263.

[18]. H. Hossam, M. M. Ghantous and M. A. -M. Salem, "Camera-based Human Counting for COVID-19 Capacity Restriction," 2022 5th International Conference on Computing and Informatics (ICCI), New Cairo, Cairo, Egypt, 2022,

[19]. Luo, Z., Fang, Z., Zheng, S., Wang, Y. and Fu, Y., 2021, August. NMS-loss: Learning with non-maximum suppression for crowded pedestrian detection. In Proceedings of the 2021 International Conference on Multimedia Retrieval.

