

Article

# Semantic Scene Understanding with Large Language Models on Unmanned Aerial Vehicles

J. de Curtò <sup>1,2,3,4,\*</sup> , I. de Zarzà <sup>1,2,3,4</sup>  and Carlos T. Calafate <sup>2</sup> <sup>1</sup> Centre for Intelligent Multidimensional Data Analysis, HK Science Park, Shatin, Hong Kong<sup>2</sup> Departamento de Informática de Sistemas y Computadores, Universitat Politècnica de València, 46022 València, Spain<sup>3</sup> Informatik und Mathematik, GOETHE-University Frankfurt am Main, 60323 Frankfurt am Main, Germany<sup>4</sup> Estudis d'Informàtica, Multimèdia i Telecomunicació, Universitat Oberta de Catalunya, 08018 Barcelona, Spain

\* Correspondence: decurto@em.uni-frankfurt.de

**Abstract:** Unmanned Aerial Vehicles (UAVs) are able to provide instantaneous visual cues and a high-level data throughput that could be further leveraged to address complex tasks, such as semantically rich scene understanding. In this work, we built on the use of Large Language Models (LLMs) and Visual Language Models (VLMs), together with a state-of-the-art detection pipeline, to provide thorough zero-shot UAV scene literary text descriptions. The generated texts achieve a GUNNING Fog median grade level in the range of 7–12. Applications of this framework could be found in the filming industry and could enhance user experience in theme parks or in the advertisement sector. We demonstrate a low-cost highly efficient state-of-the-art practical implementation of microdrones in a well-controlled and challenging setting, in addition to proposing the use of standardized readability metrics to assess LLM-enhanced descriptions.

**Keywords:** scene understanding; large language models; visual language models; CLIP; GPT-3; YOLOv7; UAV



**Citation:** de Curtò, J.; de Zarzà, I.; Calafate, C.T. Semantic Scene Understanding with Large Language Models on Unmanned Aerial Vehicles. *Drones* **2023**, *7*, 114. <https://doi.org/10.3390/drones7020114>

Academic Editors: Diego González-Aguilera and Federico Tombari

Received: 16 December 2022

Revised: 31 January 2023

Accepted: 6 February 2023

Published: 8 February 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction and Motivation

Unmanned Aerial Vehicles (UAVs) have proven to be an essential asset for practically addressing many challenges in vision and robotics. From surveillance and disaster response to the monitoring of satellite communications, UAVs perform well in situations where seamless mobility and high-definition visual capture are necessary. In this work, we focused on tasks that require a semantic understanding of visual cues and that could guide initial estimates in proposing an adequate characterization of a certain environment. Problems that are of interest include semi-adaptive filming [1] and automatic literary text description. In this setting, we propose a complete pipeline that provides real-time original text descriptions of incoming frames or a general scene description given some pre-recorded videos. The descriptions are well-suited to creating an automatic storytelling framework that can be used in theme parks or family trips alike.

Foundation models are techniques based on neural networks that are trained on large amounts of data and that present good generalization capabilities across tasks. In particular, Natural Language Processing (NLP) has seen a dramatic improvement with the appearance of GPT-2 [2] and its subsequent improvements (GPT-3 [3]). Indeed, Large Language Models (LLMs) and Visual Language Models (VLMs) have recently arisen as a resource for determining widespread problems in disciplines from robotics manipulation and navigation to literary text description, completion, and question answering. We attempt to introduce these techniques in the field of UAVs by providing the vehicle with enhanced semantic understanding. Our approach uses a captioning technique based on CLIP [4,5],

along with the YOLOv7 detector [6], which enhances the captioning output with the object annotations detected and then wires the text into GPT-3.

The descriptions provided are accurate and show a detailed understanding of the scene, and they introduce hallucinated elements that yield sound and consistent seed captions. The literary style allows for the system to be used in a wide variety of situations; for example, a human companion can use the generated text for assistance in writing a script.

The system can be used without fine-tuning in a wide variety of environments, as the base models are trained on large amounts of data. However, to further improve the consistency of the descriptive text, a proper fine-tuning of the detector could be useful when the objects that the system would normally encounter are not present in the COCO object classes [7,8], or when one wants to emphasize certain aspects of the visual cues; for instance, in an amusement park, a fine-tuning of the data could add specificity to the descriptions, e.g., providing captions that include trademark imaginary characters or specific attractions, rides, or games.

This article proposes, from the point of view of system integration, a novel zero-shot literary text description system using state-of-the-art large-language modules through the use of microdrones (RYZE Tello) and UAVs; additionally, a proposed set of measures is newly introduced in this context to assess the adequacy of the output text for the target audience.

One of the main technical issues of applying LLMs to UAVs is that the data have to be relayed to the computer, where either computation has to take place or a query has to be formulated to use an API. On-board processing is possible, but it is limited due to the amount of GPU memory that state-of-the-art models need. A high-definition camera, well-calibrated and possibly stabilized, is crucial for the optimal behavior of the overall system, as it mainly relies on visual cues for processing the entire pipeline. Another limitation is due to the object detector (YOLOv7) that is used to improve the query formulation prior to using GPT-3; in this particular setting, we used a pretrained model trained on the COCO dataset, but specific training data may be needed for a target application. Furthermore, the object detector could be integrated into the on-board processing using a CORAL board.

The main goal of this manuscript is to propose a system that could be used in many real-life applications. The majority of the techniques used have been thoroughly tested in standard datasets before, but there has been little experimentation in real settings with varying conditions and equipment. For testing the system, we used standardized measures originally used to assess texts written by human instructors in the context of the military, education, and so on.

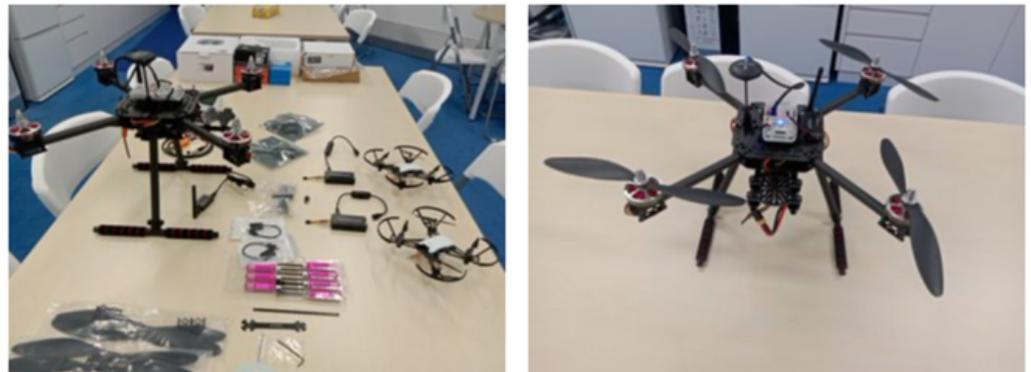
## 2. Contribution and Paper Organization

A low-cost, highly efficient practical implementation of the system was performed through the use of microdrones (e.g., RYZE Tello), which perform real-time video streaming on a ground computer that controls the vehicle. The level of autonomy of the system could be further enhanced by performing part of the computation on-device; for example, by the attachment of a CORAL Dev Board Mini (Google), which only adds 26 g of payload, to the body of the microdrone. This endows the UAV with a TPU (2GB) that can process on-device real-time detections, for instance, through the use of state-of-the-art models such as SSD MobileNet V2 [9] and EfficientDet-Lite3x [10].

The RYZE Tello drone is a compact and lightweight quadrotor drone designed for use in educational and recreational applications. It is equipped with an Intel processor and a variety of sensors, including a camera, an IMU, and ultrasonic range finders. The drone is capable of autonomous flight using a pre-programmed set of commands and can be controlled remotely using a compatible device, such as a smartphone. It is also equipped with a number of interactive features, such as gesture control and throw-to-fly, which allow users to easily interact with the drone in a variety of ways; that is, the RYZE Tello drone is

a versatile and user-friendly platform that is well-suited for a wide range of applications, including education, entertainment, and research.

A more professionally driven, inexpensive prototype appropriate for outdoor use was attempted by the use of an NXP Hover Games Drone Kit with a CORAL Dev Board Mini (Google) and a high-definition camera (see Figure 1). It also includes GPS and a Flight Management Unit (FMU) that supports the PX4 autopilot flight stack. Autonomy could be enhanced by the use of a LiDAR lite-v3 for navigation purposes, a lightweight 23 g light-ranging device with a high accuracy and range (40 m). In a well-controlled situation, such as a film studio, a tethered UAV could be used to eliminate the limitation of the battery capacity of the vehicle.



**Figure 1.** RYZE Tello Microdrones and the NXP Hover Games Drone Kit.

The NXP Hover Games Drone Kit is a hardware and software platform designed for the development and evaluation of autonomous drone systems. It includes a quadrotor drone equipped with an NXP S32 processor, a variety of sensors including an IMU, ultrasonic range finders, and stereo cameras, and a range of peripherals such as LED lights and a buzzer. The kit also includes a software library and sample code for implementing various autonomous flight behaviors such as hovering, takeoff, and landing. It is intended for use by researchers and developers working in the field of autonomous drone systems, and can be used for a wide range of applications, including drone racing, search and rescue, and aerial photography. Overall, the NXP Hover Games Drone Kit is a comprehensive and versatile tool for exploring the capabilities and limitations of autonomous drone systems.

Experimental results based on a UAV testbed show that the proposed pipeline is able to generate accurate state-of-the-art zero-shot UAV literary text descriptions.

The remainder of the paper is structured as follows: an overview of state-of-the-art approaches that entail the use of foundation models is provided. Next, Section 4 addresses the proposed methodology, as well as the background for the prior knowledge needed for the experimental assumptions, while experiments are presented in Section 5. Section 6 proposes standardized readability metrics to evaluate LLM-generated descriptions. Finally, Section 7 provides the conclusions and describes further work.

### 3. Overview and State of the Art

Large Language Models (LLMs) [11–13] and Visual Language Models (VLMs) [5] have emerged as an indispensable resource to characterize complex tasks and bestow intelligent systems with the capacity to interact with humans in an unprecedented way. These models, also called foundation models, are able to perform well in a wide variety of tasks, e.g., in robotics manipulation [14–16], and can be wired to other modules to act robustly in highly complex situations, such as in navigation and guidance [17,18].

LLMs are ML models that are trained on very large datasets of text and are capable of generating human-like text. These models are typically based on neural networks, which are composed of interconnected processing units that are able to learn and adapt through training. The goal of large language models is to learn the statistical patterns and

relationships present in the training data and use this knowledge to generate coherent and plausible text.

One of the key features of large language models is their ability to generate text that is difficult to distinguish from text written by humans. These models are trained on vast amounts of text and, as a result, are able to capture a wide range of linguistic patterns and structures, including syntax, grammar, and vocabulary. This enables them to generate text that is highly coherent and grammatically correct, and these models can thus be used for a variety of tasks, such as translation, summarization, and text generation.

In addition to their language generation capabilities, large language models have also been shown to be effective at a variety of natural language processing tasks, including language translation, question answering, and text classification. In essence, LLMs are a powerful and versatile tool for understanding and working with natural language data.

Visual Language Models (VLMs) are ML models that are trained on large datasets of text and images and are capable of generating natural language text that is coherent and grammatically correct. The goal of VLMs is to learn the statistical patterns and relationships present in the training data and use this knowledge to generate text that is descriptive and informative about the visual content of an image or a set of images.

One of the key features of visual language models is their ability to generate text that is grounded in the visual content of an image or a set of images. This means that the text generated by these models is specifically related to the objects, people, and events depicted in the image and provides descriptive and informative details about these elements. For example, a VLM could be used to generate a caption for an image depicting the occurrence of a particular action.

In addition to generating descriptive text, visual language models can also be used for a variety of other tasks, such as image classification, object detection, and image captioning. These models can be trained to recognize and classify different types of objects and events in an image and can also be used to generate coherent and grammatically correct captions that describe the content of an image.

VLMs are a powerful and versatile tool for understanding and working with both text and image data. By enabling the generation of descriptive and informative text that is grounded in the visual content of an image, these models have the potential to facilitate a wide range of applications, including image and video analysis, content generation, and natural language processing.

Drones, also known as unmanned aerial vehicles (UAVs), have the potential to be used for a wide range of applications involving semantic scene understanding, which refers to the ability of a system to analyze and interpret the meaning or significance of the objects, people, and events present in a scene. This capability is important for many applications, including robotics, surveillance, and autonomous driving.

One way in which drones can be used for this particular purpose is through the use of on-board sensors and cameras to capture visual data and other types of data about the environment. These data can then be processed and analyzed using ML algorithms to identify and classify the objects and events present in the scene. For example, a drone equipped with a camera and an object recognition algorithm could be used to identify and classify different types of objects in a scene, such as vehicles, pedestrians, and buildings.

In addition to object recognition, drones can also be used for other types of tasks, such as event detection and tracking. For example, a drone equipped with a camera and an event detection algorithm could be used to identify and track the movements of people or vehicles in a scene. This could be useful for applications such as surveillance or traffic monitoring. By enabling the analysis and interpretation of the meaning or significance of objects and events in a scene, drones can provide valuable insights and information for a variety of tasks and scenarios. In this work, we built on the improvements in object detection [19,20] and model reparameterization [21,22] to apply LLMs and VLMs in the field of Unmanned Aerial Vehicles (UAVs) [1]. State-of-the-art techniques of captioning [23–25] have allowed computers

to semantically understand visual data, while advances in automated storytelling can now generate realistic storylines from visual cues [26,27].

#### 4. Methodology

UAV real-time literary storytelling refers to the use of Unmanned Aerial Vehicles (UAVs), also known as drones, to generate narrative stories in real-time based on data they collect. This could involve using the UAVs to capture visual data and other types of data about the environment and then processing and analyzing these data using ML algorithms to identify the objects and events present in the scene. The resulting data could then be used to generate a narrative story that describes and explains the objects and events in the scene coherently and grammatically.

One potential application of UAV real-time literary storytelling is in the field of journalism, where UAVs could be used to capture newsworthy events and generate narratives about these events in real time. For example, a UAV could be used to capture images and video of a natural disaster and then generate a narrative story about the disaster that is based on the data collected by the UAV. This could provide a more immersive and interactive way of reporting on events and could enable journalists to generate stories more quickly and efficiently.

Another potential application is in the field of entertainment, where UAVs could be used to capture data about live events and generate interactive narratives about these events in real time. For example, a UAV could be used to capture data about a sports game and then generate a narrative story about the game that is based on the data collected by the UAV. This could provide a more engaging and interactive way of experiencing live events and could enable users to experience events in a more immersive and interactive way.

UAV real-time literary storytelling offers potential for a wide range of applications, including journalism, entertainment, and education. By enabling the generation of narrative stories in real time based on data collected by UAVs, this technology has the potential to facilitate a more immersive and interactive way of experiencing and understanding events and situations.

CLIP (Contrastive Language-Image Pre-training) is a neural network architecture developed by researchers at OpenAI that can be used for image captioning and other natural language processing tasks. It is based on the idea of pre-training a model on a large dataset of images and text and then fine-tuning it for a specific task, such as image captioning.

CLIP uses a transformer architecture, which is a type of neural network that is particularly well-suited for tasks involving sequential data, such as natural language processing. The model is trained to predict the next word in a sentence given the previous words, using the images as additional context. One key feature of CLIP is that it is able to learn a continuous space of image and text representations, which allows it to generate high-quality captions for a wide range of images. It is also able to learn from a large amount of data, which helps it to generalize to new images and improve the performance in the image captioning task.

The problem of captioning can be formulated as follows: given a dataset of paired images and captions  $\{x^z, c^z\}_{z=1}^N$ , the aim is to be able to synthesize adequate captions given an unseen sample image. In our approach, we built on recent work that uses the embedding of CLIP as a prefix to the caption and that is based on the next objective, where the captions can be understood as a sequence of tokens  $c^z = c_1^z, \dots, c_\ell^z$ , padded to a maximum length  $\ell$ :

$$\max_{\theta} \sum_{z=1}^N \sum_{w=1}^{\ell} \log p_{\theta}(c_w^z | x^z, c_1^z, \dots, c_{w-1}^z). \quad (1)$$

We consider, as in [4], an autoregressive language model that predicts the consequent token without considering future tokens.

The CLIP embedding is then projected by a mapping network, denoted as  $F$ :

$$p_1^z, \dots, p_k^z = F(\text{CLIP}(x^z)). \quad (2)$$

where  $p_w^z$  is a vector with the same dimension as a word embedding and then concatenated with the caption embedding. A cross-entropy loss is used to train the mapping  $F$ .

YOLO (You Only Look Once) [19,20] is a real-time object detection algorithm. It is an end-to-end neural network model that is able to detect and classify objects in images and videos. YOLO works by dividing the input image into a grid of cells and predicting the class and location of objects within each cell. The model uses anchor boxes to make predictions at multiple scales, so it can detect objects of different sizes. The model also predicts the confidence of each detection, which helps to filter out false positives.

One of the main advantages of YOLO is its speed. It is able to process images and videos in real time, making it suitable for use in applications such as video surveillance and autonomous vehicles. YOLO has undergone several versions, with each version improving the accuracy and efficiency of the model. YOLOv7 is the latest version of YOLO and includes several enhancements over previous versions.

We propose a general pipeline for UAV real-time literary storytelling (see Figure 2) that is based on the previously described captioning technique that utilizes CLIP prefix captioning [4,5,28] and that combines the obtained sentence trained with Conceptual Captions [29] with detections given by YOLOv7 [6]. The output of the object detector is processed by a module of sentence formation such that it can be fed into a GPT-3 module, which provides an enhanced literary description. A query formulating the task to be determined by GPT-3 is needed. The system can work in real time on the streaming frames of the vehicle or as a post-processing module once the UAV has landed.

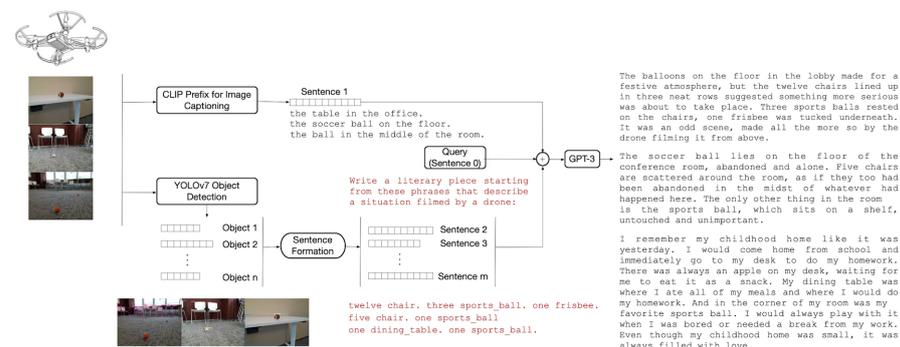


Figure 2. UAV real-time literary storytelling.

The pipeline does not require fine-tuning to specific tasks, although it would benefit from such tuning if used in a particular environment where some specific objects need to be identified, e.g., when there is a need to be specific in terms of trademark names.

The main blocks of the architecture are CLIP Prefix for Image Captioning, YOLOv7, and GPT-3.

CLIP Prefix for Image Captioning is a transformer-based architecture that enables the generation of captions while the CLIP and GPT-2 model are frozen. It consists of the training of a lightweight mapping network based on a transformer [30,31] that translates from the CLIP embedding space to GPT-2.

YOLOv7 is the state-of-the-art object detector in terms of speed and accuracy. It is a generalization of previous YOLO-based architectures with the use of Extended Efficient Layer Aggregation Networks (E-ELANs) [6]. E-ELANs address the problem of controlling the shortest longest gradient path so that the network converges effectively. It uses expand, shuffle, and merge cardinality to continue learning without losing the original gradient path.

GPT-3 is used to enhance the captions by the natural language instruction and prompt engineering. All of our experiments used the API of OpenAI, and the model is surprisingly effective with zero-shot prompts.

Having said that, the manuscript has the goal of deploying state-of-the-art LLMs to accomplish the task of zero-shot semantic scene understanding through the use of a low-cost UAV (RYZE Tello or a NXP Hover Games Drone Kit) that incorporates a high-definition camera. Further integration by the use of a Raspberry Pi Zero W or a CORAL board can move some of the computation on-device with the proper module adaptation, both for object detection and also for the LLM API. In the latter case, a call to OpenAI API is necessary at this stage but advances on the field will soon make it possible to test the trained models directly on-board (e.g., pruning the LLM model to make it fit on memory) without the need to relay the video frames to the computer for further processing. In either way, model pruning can be used to reduce the model size and thus reduce the computational requirements. Another technique would be to use model quantization to reduce the precision of the model and make it more efficient. Additionally, another viable approach is knowledge distillation, where the knowledge of a large teacher model is transferred to a smaller student model for the purpose of using it on a resource-constrained environment.

## 5. Results and Experiment Set-Up

Experiments were conducted on a well-controlled challenging environment with the use of RYZE Tello, streaming the data in real time to a ground computer that processes the frames one by one. Figures 3–6 illustrate all of the stages of the used methodology for a number of UAV captured stream frames, with contrasting levels of descriptive goodness. The drone captures a particular visual scene that is consequently sent to the ground computer, where a first caption is generated using CLIP Prefix for Image Captioning with beam search. The caption is improved by the output of a YOLOv7 object detector after sentence formation. Finally, a query is formulated together with the resultant caption to generate an enhanced text description by the GPT-3 module. The results are consistent and robust and exhibit original and genuine descriptions of the imagery, and the modules of captioning and object detection are effective and efficient.

CLIP Prefix for Image Captioning presents an average runtime of 206 ms and the YOLOv7 Object Detector presents one of 556 ms, using a GPU Tesla V100-SXM2 (16GB) and a two-core Intel(R) Xeon(R) CPU @ 2.00 GHz. Prompts to the OpenAI API take an average of 3.9 s using the completion model text-davinci-002, where max\_tokens = 401 and temperature = 0.9.

Figure 3 shows the methodology when the CLIP captioning module and the YOLOv7 object detection produce accurate outputs, and the GPT-3 module produces a very good enhanced literary description.

Figure 4 shows the methodology when the CLIP captioning module and the YOLOv7 object detection produce relatively good outputs (detection of the majority of objects—not all or not completely accurate), and the GPT-3 generates a realistic literary description but with the presence of hallucinated elements that provide realism but are not actually in the scene.

Figure 5 shows the methodology when the CLIP captioning module and the YOLOv7 produce somewhat adequate outputs, but they are not particularly accurate, e.g., detecting objects but misclassifying some of them, or generating overly general caption descriptions. The GPT-3 then produces an enhanced description, but not a very accurate one.

Finally, Figure 6 shows the methodology when the CLIP captioning module or YOLOv7 object detection fail to describe the scene accurately, and the GPT-3 module generates an erroneous text description.

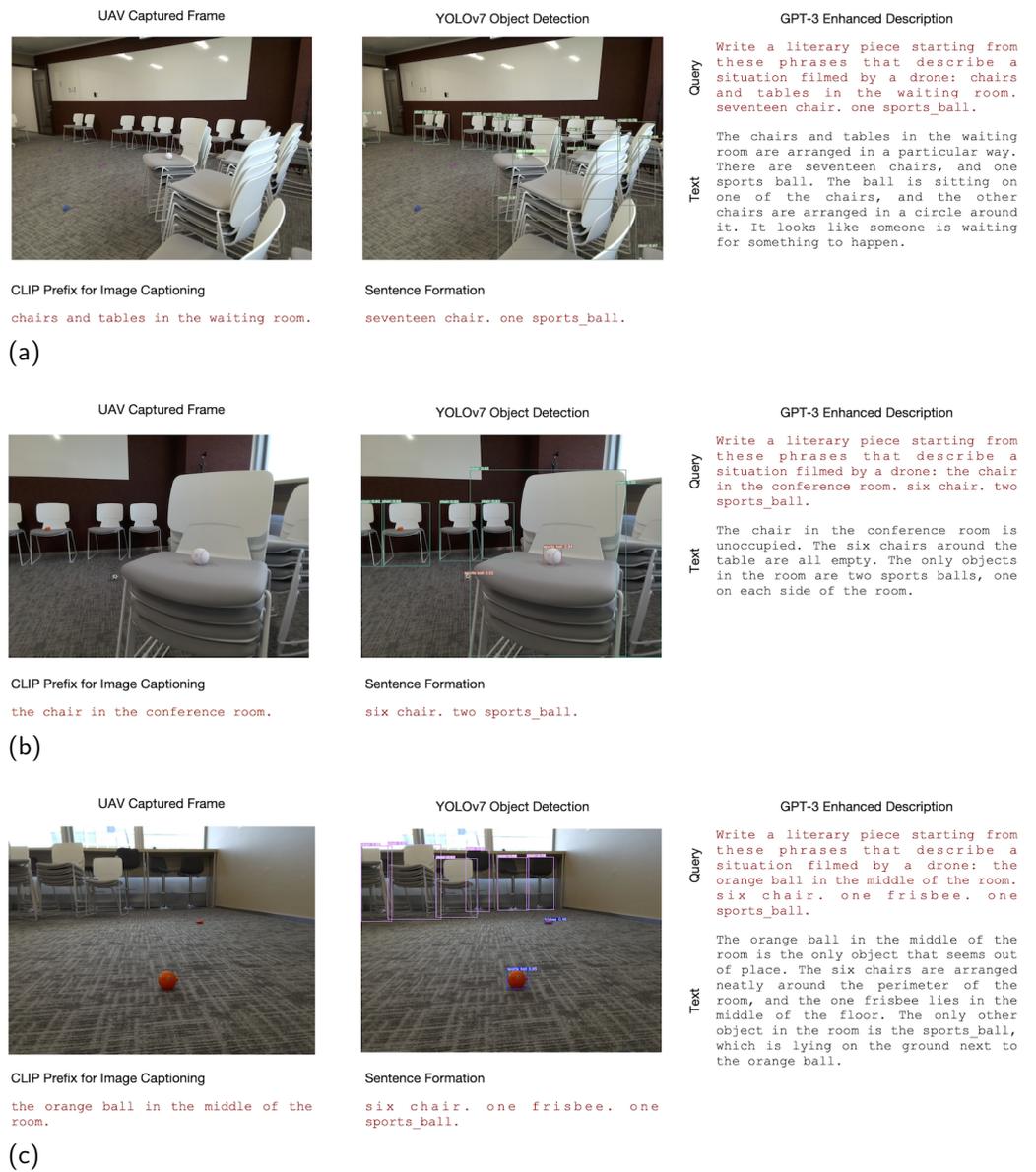
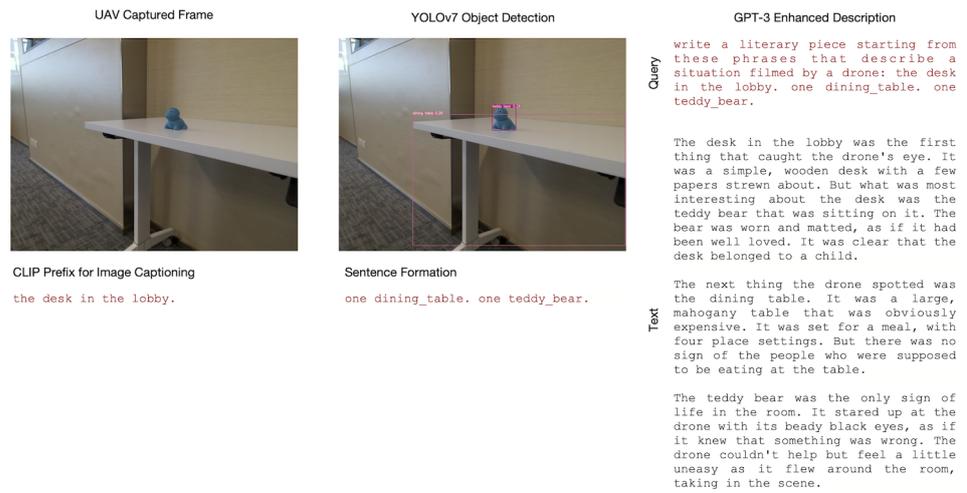


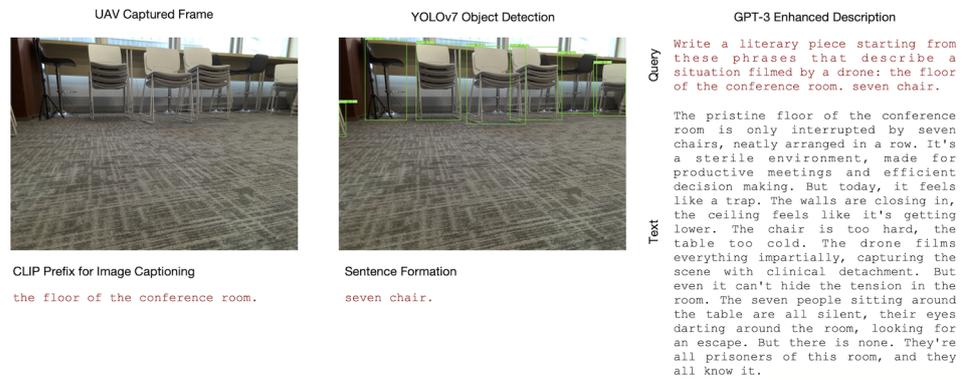
Figure 3. UAV captured frame processing and GPT-3. Very good GPT-3 descriptions of the scene.



Figure 4. Cont.



(b)



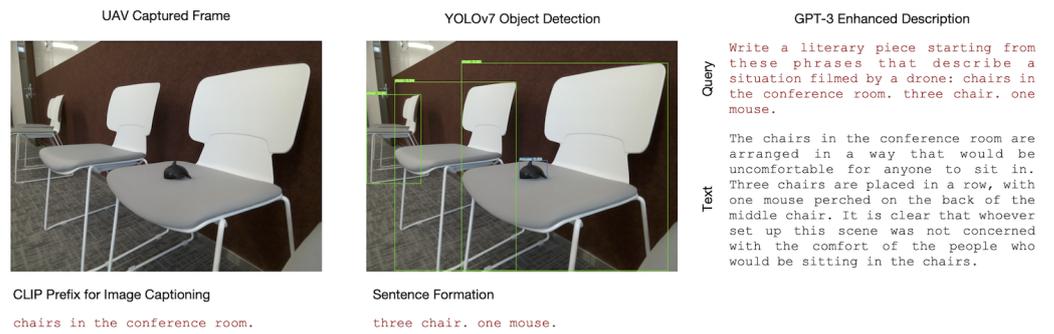
(c)

Figure 4. UAV captured frame processing and GPT-3. Adequate literary GPT-3 descriptions.



(a)

Figure 5. Cont.

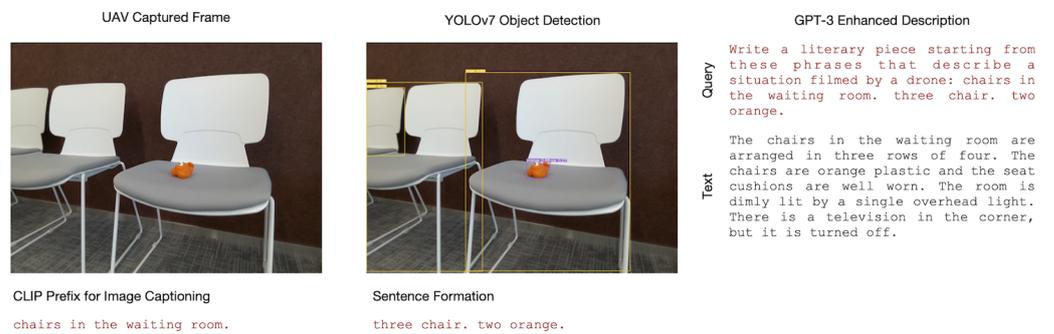


(b)

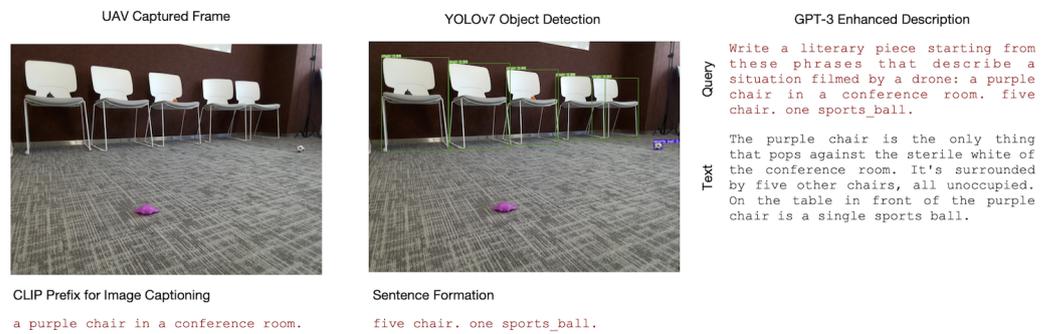


(c)

**Figure 5.** UAV captured frame processing and GPT-3. Somewhat good descriptions, but the CLIP captioning module and the YOLOv7 produce inaccurate outputs.



(a)



(b)

**Figure 6.** UAV captured frame processing and GPT-3. Failure cases.

### 6. Readability Analysis

GPT-3 (short for “Generative Pre-training Transformer 3”) is a large language model developed by OpenAI that is trained on a very large dataset of text and is capable of

generating human-like text. It is based on a type of neural network called a transformer, which is composed of interconnected processing units that are able to learn and adapt through training. The goal of GPT-3 is to learn the statistical patterns and relationships present in the training data and use this knowledge to generate coherent and plausible text.

One of the key features of GPT-3 is its ability to generate text that is difficult to distinguish from text written by humans. It is trained on a dataset of billions of words and, as a result, is able to capture a wide range of linguistic patterns and structures, including syntax, grammar, and vocabulary. This enables it to generate text that is highly coherent and grammatically correct, and it can thus be used for a variety of tasks, such as translation, summarization, and text generation.

Readability measures are tools that are used to evaluate the complexity of written text and determine how easy or difficult it is for readers to understand. One common readability measure, for instance, is the GUNNING Fog index, which is a formula that estimates the number of years of education a reader would need to understand a piece of text. The GUNNING Fog index is based on the average number of words per sentence and the percentage of complex words (those with three or more syllables) in the text.

To calculate the GUNNING Fog index, the following steps are followed:

- Count the number of words in a sample of the text;
- Count the number of sentences in the sample;
- Divide the total number of words by the total number of sentences to calculate the average number of words per sentence;
- Count the number of complex words (those with three or more syllables) in the sample;
- Divide the number of complex words by the total number of words, and multiply the result by 100 to calculate the percentage of complex words in the sample;
- Add the average number of words per sentence and the percentage of complex words. The result is the GUNNING Fog index.

The GUNNING Fog index is typically used to evaluate the readability of written materials, such as reports, documents, and articles. It is a useful tool for determining the level of difficulty of a piece of text and ensuring that it is appropriate for a particular audience. For example, a text with a GUNNING Fog index of 8 would be considered suitable for readers with an eighth-grade education or higher.

Such readability measures are useful tools for evaluating the complexity of written text and ensuring that it is appropriate for a particular audience. This can help writers and editors to produce written materials that are clear, concise, and easy to understand and can help readers to more easily comprehend and retain information presented in a text.

A readability analysis of the GPT-3-enhanced text is provided by the use of standardized measures, the one introduced earlier being the most effective. In this manuscript, we propose analyzing LLM texts by the following metrics: FLESCH reading ease, DALE CHALL readability, the Automated Readability Index (ARI), the COLEMAN LIAU index, GUNNING Fog, SPACHE, and Linsear Write. The scores obtained by the use of these formulas were designed by linguists to assess the readability of texts to approximate their usability and have been extensively used by, for example, the Office of Education of the United States of America to calibrate the readability of textbooks for the public school system, daily newspapers and monthly magazines to target the appropriate audience, the Department of Defense to help assess the adequacy of technical manuals, and, in general, many US Government Agencies to evaluate the difficulty of a reading passage written in English.

FLESCH reading ease [32] is a simple approach used to assess the grade level of the reader. It is based on the average sentence length and the average number of syllables per word. It is a score in the set  $[0, 100]$ ; the higher the number, the easier the text is to read. According to the scale,  $[0, 30]$  means a text is easily understood by a college graduate,  $[60, 70]$  means it is easily understood by eighth and ninth graders, and  $[90, 100]$  means it is easily understood by a fifth grader.

DALE CHALL readability [33] calculates the grade level of a text sample based on the average sentence length in words and the number of difficult words according to a designated list of common words familiar to most fourth-grade students. Adjusted scores are as follows: <5: Grade 4 and below; [5, 6): Grades 5–6; [6, 7): Grades 7–8; [7, 8): Grades 9–10; [8, 9): Grades 11–12; [9, 10): College;  $\geq 10$ : College Graduate.

The Automated Readability Index (ARI) consists of a weighted sum of two ratio factors: the number of characters per word, and the average number of words per sentence. It assesses the understandability of a text and outputs a value that approximates the grade level needed to grasp the text. For example, the tenth grade corresponds to 15–16 years old, the eleventh grade corresponds to 16–17 years old, the twelfth grade corresponds to 17–18 years old, and greater than twelve corresponds to the level of college.

The COLEMAN-LIAU index [34] is similarly based on the average number of letters per 100 words and the average number of sentences per 100 words. It is like the ARI, but unlike most of the other metrics that predict the grade level, it relies on characters instead of syllables per word.

GUNNING Fog [35] is based on the scaled sum of the average sentence length and the percentage of hard words. It measures the readability of a text passage, and the ideal value is 7 or 8. Texts with a score above 12 are too hard for most people to understand. The measure scores highly with short sentences written in simple language but penalizes long sentences with complicated words.

The SPACHE readability formula [36] is based on the average sentence length and the number of difficult words according to a third grader. It is similar to Dale Chall, but for primary texts until the third grade. To assess the readability of a text, SPACHE is first used, and if the result is higher than third grade, Dale Chall is used.

Linsear Write is a readability formula based on sentence length and the number of words with three or more syllables. Analogous to the previous formulations, it scores a text passage according to the grade level.

Table 1 shows the proposed metrics on several example frames. The metrics are computed on unique frames in Row 1–3 and on multi-frame configurations in Row 4–8. We can observe that the storylines generated exhibit a relatively consistent behavior among the statistical indices, where unique frames tend to be ranked at a lower grade level and multi-frame configurations are closer to college level. All SPACHE readability indices are higher than third grade, so Dale Chall has to be considered, where the frames are consistently ranked with a median grade level of [7, 8]. Among the measures, GUNNING Fog presents an ideal behavior, as all values are in the range of [7–12], which means that the level of generated texts is comparable to that of established publications in magazines and books, and therefore can be understood by the general public while presenting a rich vocabulary.

**Table 1.** Readability analysis of a random stream of data captured by RYZE Tello. Score (upper row) and grade level (lower row) for each metric.

Frame(s)	Metric	FLESCH Reading Ease	Dale Chall	ARI	Coleman Liau	GUNNING Fog	SPACHE	Linsear Write
00		68.36 [8, 9]	6.56 [7, 8]	6.29 [7]	9.10 [9]	9.47 [9]	4.56 [5]	6.1 [6]
01		84.22 [6]	6.06 [7, 8]	3.63 [9, 10]	4.36 [4]	8.18 [8]	3.91 [4]	7.14 [7]
02		84.57 [6]	5.67 [5, 6]	3.80 [9, 10]	5.04 [5]	7.40 [7]	4.18 [4]	6.46 [6]
03–05		71.11 [7]	6.82 [7, 8]	10.27 [16, 17]	7.89 [8]	11.81 [12]	5.82 [6]	13.14 [13]
06–07		82.08 [7, 8]	6.82 [5]	4.36 [8]	7.89 [8]	7.56 [4]	3.64 [7]	6.94
08–10		74.30 [7]	6.35 [7, 8]	7.05 [13, 14]	7.53 [8]	10.58 [11]	4.87 [5]	9.0 [9]
11–13		75.94 [7]	6.33 [7, 8]	8.54 [9]	7.47 [7]	10.76 [11]	5.33 [5]	11.5 [12]

## 7. Conclusions

An RIZE Tello drone is a small, lightweight, and low-cost quadrotor drone that is equipped with a camera and is capable of autonomous flight. In this system, the drone is used to capture video footage of a scene and transmit it to a ground computer in real time.

On the ground computer, the video stream is processed using state-of-the-art LLMs together with a module of object detection to produce accurate text descriptions of a scene in the form of captions. These captions can be used to provide a verbal description of the scene for individuals who are deaf or hard of hearing, or to provide additional context for individuals who are able to see the video footage.

A pipeline for semantic scene understanding given a stream of UAV data frames was proposed. The methodology does not require fine-tuning; rather, it provides zero-shot text descriptions. The modules consist of state-of-the-art architectures. A captioning module based on CLIP Prefix for Image Captioning is wired through sentence formation to a YOLOv7 object detector, and the generated text is enhanced by prompting GPT-3 natural language instructions. We are the first to provide zero-shot UAV literary storytelling that can stream to a ground computer in real time or after landing (in this latter case, the video would be stored on an SD card, and the RYZE Tello drone needs to be equipped with a board computer, e.g., a Raspberry Pi Zero W or a CORAL board) and that provides state-of-the-art accurate literary text descriptions. Metrics used to assess the readability of LLM texts are proposed, leveraging standardized measures from linguistics.

The system combines the capabilities of an RIZE Tello drone (or an NXP Hover Games Drone) with advanced techniques of computer vision to provide a rich and detailed description of a scene in real time. The system has potential applications in a wide range of fields, including surveillance, search and rescue, and environmental monitoring.

As further work, the trajectory of the drone could be optimized for a certain filming style to help the text description module to obtain better shots for particularly interesting events that need to be addressed in the storyline. That being said, in the current work, we did not take planning and trajectory issues into consideration and assumed that the UAV is being remotely controlled or is flying using an adequate autopilot policy. In addition, GPS coordinates and positioning information from other sensors such as IMU or LiDAR could be used to further improve the resultant text descriptions by prompting the GPT-3 module with the corresponding trajectories.

There are a number of other ways that the previously described system could be extended or improved upon. Some potential areas of further work include the following.

The accuracy and reliability of the algorithms that handle captioning and object detection can be improved: while current LLMs and object detection algorithms are highly accurate, there is always room for improvement. Further research could focus on developing new techniques or fine-tuning existing algorithms to increase their accuracy and reliability. Other sensors can be added: the RIZE Tello drone is equipped with a camera, but additional sensors, such as LiDAR or RADAR, could allow the system to gather more detailed and comprehensive data about the scene. The drone's autonomy could be enhanced: the RIZE Tello drone is capable of autonomous flight, but further work could focus on developing more advanced autonomy algorithms to enable the drone to navigate more complex environments and perform more sophisticated tasks. Real-time analysis could be implemented: at the moment, the system processes the video stream and generates captions and object detections after the fact. However, implementing real-time analysis could allow the system to provide updates and alerts in near-real time, making it more useful for applications such as surveillance or search and rescue. Finally, applications could be developed for specific domains: the system could be tailored to specific domains by training the captioning and object detection algorithms on domain-specific data and developing domain-specific applications. For example, the system could be used for agricultural monitoring by training the algorithms on data specific to crops and farm machinery.

The ultimate goal is to be able to confer autonomous systems (e.g., UAVs and self-driving cars) with literary capabilities comparable to those provided by human counterparts.

Specifically, the use of LLMs and VLMs push the boundaries of system perception and the understandability of events, situations, and contextual information.

**Author Contributions:** Conceptualization, J.d.C. and I.d.Z.; funding acquisition, C.T.C.; investigation, J.d.C. and I.d.Z.; methodology, J.d.C. and I.d.Z.; software, J.d.C. and I.d.Z.; supervision, C.T.C.; writing—original draft, J.d.C.; writing—review and editing, C.T.C., J.d.C. and I.d.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work is supported by the HK Innovation and Technology Commission (InnoHK Project CIMDA). We acknowledge the support of Universitat Politècnica de València; R&D project PID2021-122580NB-I00, funded by MCIN/AEI/10.13039/501100011033 and ERDF.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare that they have no conflict of interest. The funders had no role in the design of the study, in the collection, analyses, or interpretation of data, in the writing of the manuscript, or in the decision to publish the results.

### Abbreviations

The following abbreviations are used in this manuscript:

ML	Machine Learning
NLP	Natural Language Processing
LLM	Large Language Models
VLM	Visual Language Models
GPT	Generative Pre-training Transformer
CLIP	Contrastive Language-Image Pre-training
YOLO	You Only Look Once
LiDAR	Light Detection And Ranging
RADAR	Radio Detection And Ranging

### References

- Bonatti, R.; Bucker, A.; Scherer, S.; Mukadam, M.; Hodgins, J. Batteries, camera, action! learning a semantic control space for expressive robot cinematography. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021.
- Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. *Language Models Are Unsupervised Multitask Learners*; Technical Report; 2019. Available online: <https://www.semanticscholar.org/paper/Language-Models-are-Unsupervised-Multitask-Learners-Radford-Wu/9405cc0d6169988371b2755e573cc28650d14dfe> (accessed on 15 December 2022).
- Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. *Neural Inf. Process. Syst.* **2020**, *33*, 1877–1901.
- Mokady, R.; Hertz, A.; Bermano, A.H. ClipCap: CLIP prefix for image captioning. *arXiv* **2021**, arXiv:2111.09734.
- Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning transferable visual models from natural language supervision. In Proceedings of the 38th International Conference on Machine Learning, Virtual, 18–24 July 2021.
- Wang, C.-Y.; Bochkovskiy, A.; Liao, H.-Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv* **2022**, arXiv:2207.02696.
- Chen, X.; Fang, H.; Lin, T.-Y.; Vedantam, R.; Gupta, S.; Dollár, P.; Zitnick, C.L. Microsoft coco captions: Data collection and evaluation server. *arXiv* **2015**, arXiv:1504.00325.
- Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014; Springer: Berlin/Heidelberg, Germany, 2014; pp. 740–755.
- Sandler, M.; Howard, A.G.; Zhu, M.; Zhmoginov, A.; Chen, L. Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. *arXiv* **2018**, arXiv:1801.04381.
- Tan, M.; Pang, R.; Le, Q.V. Efficientdet: Scalable and efficient object detection. *arXiv* **2020**, arXiv:1911.09070.
- Alayrac, J.-B.; Donahue, J.; Luc, P.; Miech, A.; Barr, I.; Hasson, Y.; Lenc, K.; Mensch, A.; Millican, K.; Reynolds, M.; et al. Flamingo: A visual language model for few-shot learning. *arXiv* **2022**, arXiv:2204.14198.

12. Gu, X.; Lin, T.-Y.; Kuo, W.; Cui, Y. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv* **2022**, arXiv:2104.13921.
13. Wei, J.; Tay, Y.; Bommasani, R.; Raffel, C.; Zoph, B.; Borgeaud, S.; Yogatama, D.; Bosma, M.; Zhou, D.; Metzler, D.; et al. Emergent abilities of large language models. *arXiv* **2022**, arXiv:2206.07682.
14. Cui, Y.; Niekum, S.; Gupta, A.; Kumar, V.; Rajeswaran, A. Can foundation models perform zero-shot task specification for robot manipulation? In Proceedings of the 4th Annual Learning for Dynamics and Control Conference, Stanford, CA, USA, 23–24 June 2022.
15. Nair, S.; Rajeswaran, A.; Kumar, V.; Finn, C.; Gupta, A. R3M: A universal visual representation for robot manipulation. *arXiv* **2022**, arXiv:2203.12601.
16. Zeng, A.; Florence, P.; Tompson, J.; Welker, S.; Chien, J.; Attarian, M.; Armstrong, T.; Krasin, I.; Duong, D.; Wahid, A.; et al. Transporter networks: Rearranging the visual world for robotic manipulation. *arXiv* **2022**, arXiv:2010.14406.
17. Huang, W.; Abbeel, P.; Pathak, D.; Mordatch, I. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In Proceedings of the 39th International Conference on Machine Learning, Baltimore, MA, USA, 17–23 July 2022.
18. Zeng, A.; Attarian, M.; Ichter, B.; Choromanski, K.; Wong, A.; Welker, S.; Tombari, F.; Purohit, A.; Ryoo, M.; Sindhwani, V.; et al. Socratic models: Composing zero-shot multimodal reasoning with language. *arXiv* **2022**, arXiv:2204.00598.
19. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 779–788.
20. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
21. Tan, M.; Le, Q.V. Efficientnet: Rethinking model scaling for convolutional neural networks. *arXiv* **2019**, arXiv:1905.11946.
22. Tan, M.; Le, Q.V. Efficientnetv2: Smaller models and faster training. *arXiv* **2021**, arXiv:2104.00298.
23. Anderson, P.; He, X.; Buehler, C.; Teney, D.; Johnson, M.; Gould, S.; Zhang, L. Bottom-up and top-down attention for image captioning and visual question answering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6077–6086.
24. Fang, H.; Gupta, S.; Iandola, F.; Srivastava, R.K.; Deng, L.; Dollár, P.; Gao, J.; He, X.; Mitchell, M.; Platt, J.C.; et al. From captions to visual concepts and back. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1473–1482.
25. Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; Bengio, Y. Show, attend and tell: Neural image caption generation with visual attention. In Proceedings of the 32nd International Conference on International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 2048–2057.
26. Fan, A.; Lewis, M.; Dauphin, Y. Hierarchical neural story generation. *arXiv* **2018**, arXiv:1805.04833.
27. See, A.; Pappu, A.; Saxena, R.; Yerukola, A.; Manning, C.D. Do massively pretrained language models make better storytellers? *arXiv* **2019**, arXiv:1909.10705.
28. Li, X.L.; Liang, P. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv* **2021**, arXiv:2101.00190.
29. Sharma, P.; Ding, N.; Goodman, S.; Soricut, R. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Melbourne, Australia, 15–20 July 2018; pp. 2556–2565.
30. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S. et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2021**, arXiv:2010.11929.
31. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *arXiv* **2017**, arXiv:1706.03762.
32. Flesch, R. A new readability yardstick. *J. Appl. Psychol.* **1948**, *32*, 221–233. [[CrossRef](#)]
33. Dale, E.; Chall, J.S. A formula for predicting readability. *Educ. Res. Bull.* **1948**, *27*, 11–28.
34. Coleman, M.; Liau, T.L. A computer readability formula designed for machine scoring. *J. Appl. Psychol.* **1975**, *60*, 283–284. [[CrossRef](#)]
35. Gunning, R. *The Technique of Clear Writing*; McGraw-Hill: New York, NY, USA, 1952.
36. Spache, G. A new readability formula for primary-grade reading materials. *Elem. Sch. J.* **1953**, *53*, 410–413. [[CrossRef](#)]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.