Landing Zone Detection for MAVs using Depth Images and Vision Transformers

Victoria Eugenia Vazquez-Meza and Jose Martinez-Carranza*
Instituto Nacional de Astrofisica Optica y Electronica

ABSTRACT

We present a methodology for landing zone detection in Micro Aerial Vehicles (MAVs) using Vision Transformers (ViTs). We present results on the use of ViTs due to their ability to capture spatial relations through the attention mechanism, potentially offering superior performance with fewer training examples than other Deep Neural Networks. Experiments with aerial images, from a dataset and depth images captured with a depth camera on board a drone, confirm ViT's superiority over other widely used Convolutional Network such as ResNet, particularly with limited training data. Despite the noisy depth images, captured with the depth camera, the ViT model can be used to detect landing zones with an average processing time of 11.9 ms on outdated GPU hardware.

1 INTRODUCTION

Vision Transformers (ViTs) represent a significant advancement in computer vision, leveraging the transformer architecture traditionally used in natural language processing. ViTs have demonstrated superior performance on various image classification benchmarks, often surpassing convolutional neural networks (CNNs). They handle larger datasets and more complex tasks efficiently, benefiting from the ability to learn long-range dependencies within the data. Furthermore, ViTs have been shown to learn faster with fewer examples than CNNs [1], making them suitable for tasks that require rapid prediction models. This advantage is particularly useful in scenarios, where there is limited time to collect training data and where short training times are desirable. Due to these properties, in this work, we explore the use of ViTs for landing zone detection using depth images captured with a camera on board a Micro Aerial Vehicle (MAV) that could be navigating an environment, seeking a suitable place to perform its landing.

Due to the advancements in hardware and software, today it is possible for MAVs to carry small depth cameras such as the Real-Sense or the OAK-D camera [2, 3], which provide chromatic and depth images from the scene. When depth cameras are not available, a possibility is to infer depth from chromatic images using Deep Learning [4, 5]. In both cases, we can remove the dependency on more expensive range sensors such as LiDAR or radar, which usually are bulky and expensive. Moreover, in the work of [6], it was shown that a CNN could be trained to detect more effectively landing zones using depth images as input to the network, rather than when using chromatic (RGB) images.

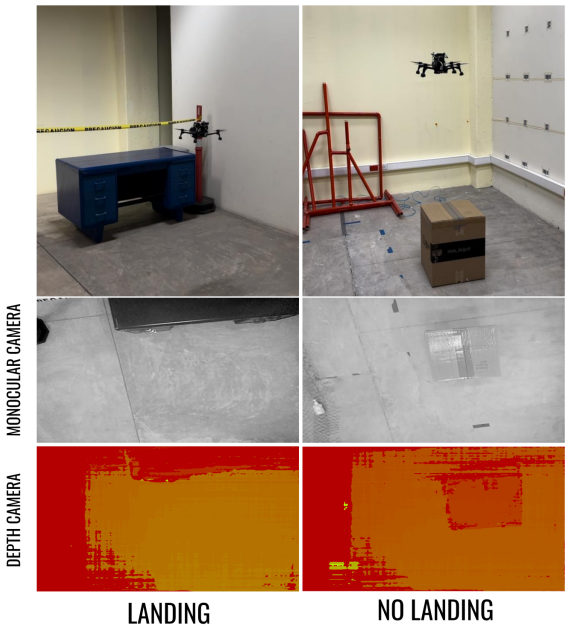

Figure 1: We leverage the attention mechanism of a Vision Transformer model to predict whether a landing zone is present in a depth image. We used the OAK-D camera onboard a MAV to obtain depth images. It is noteworthy that the chromatic image to the right barely reveals the presence of an object in the middle, which could mislead a detector based on chromatic information. In contrast, the object is more clearly visible in the depth image.

Motivated by the above, in this work, we explore the use of ViTs and compare it against using a state-of-the-art Residual Neural Network (ResNet) [7]. Our goal has been to investigate the performance of ViTs when learning to associate depth information to landing zones, whose depth maps are expected to show regular patterns in the depth values as a result of the uniformity of the terrain, which makes it good for a landing zone.

*Email address(es): victoria.vazquez@inaoep.mx, carranza@inaoep.mx

We conducted our experiments using the dataset ESPADA [5], which provides pairs of RGB and depth aerial images. We also evaluated our approach using real images captured with a drone. Our results indicate that ViTs outperform ResNet requiring even less training data, making them a robust and effective approach to address the landing zone detection problem. Similarly, we also carried out experiments using depth images captured with the OAK-D camera onboard a MAV while flying indoors (see Fig. 1, confirming that the ViTs, when trained with a small dataset, outperforms ResNet in accuracy and slightly in operation frequency.

In order to present our approach, this paper has been organised as follows: the related work is presented in Section 2; our methodology is described in Section 3; the experimental framework is presented in Section 4 and finally, our conclusion are drawn in Section 5.

2 RELATED WORK

The increasing use of autonomous MAVs has created a need for safe landing zone (SLZ) detection techniques[8, 9], particularly in scenarios where MAVs experience technical difficulties, such as low battery, adverse weather conditions, mechanical failures, and interference. In these situations, the MAVs have an urgent need to land safely. The issues surrounding MAV landing has attracted wide attention, thus resulting in the development of two main detection techniques: non-vision-based [10, 11, 12] and vision-based[13, 14, 6].

Vision-based techniques offer distinct advantages, including strong autonomy, cost-effectiveness, and robust anti-interference capabilities. These techniques encompass various approaches: *(1) Camera-based techniques*, which utilise different configurations of cameras, such as monocular, stereo ranging, or multi-camera setups.[15, 16] *(2) Structure from Motion (SfM)* involves processing a sequence of 2D images to create a 3D terrain surface, enabling the estimation of orientation, photo location, and camera parameters such as focal length, radial, and tangential distortion. *(3) Colour segmentation* is based on grayscale thresholding to identify relevant features. *(4) Simultaneous Localisation and Mapping (SLAM)* estimates MAV localisation by generating a 3D map of the environment and using it to guide the landing [17]. Currently, due to the demonstrated capabilities of deep learning, camera-based techniques can be integrated into computer vision classification algorithms for detecting landing zones in MAV applications.

Deep convolutional neural networks (DCNNs) have led to significant breakthroughs in image classification. [18, 19] DCNNs naturally integrate low, mid, and high-level features and classifiers and can extract different features due to the number of stacked layers [20]. To date, researchers have proposed several CNN architecture models, including VGG, LeNet, GoogleLeNet, and AlexNET. We selected ResNet-50 for CNN image classification due to its advantages in addressing network degradation, and compared to GoogLeNet and VGG, ResNet offers greater computational efficiency and lower time complexity [21].

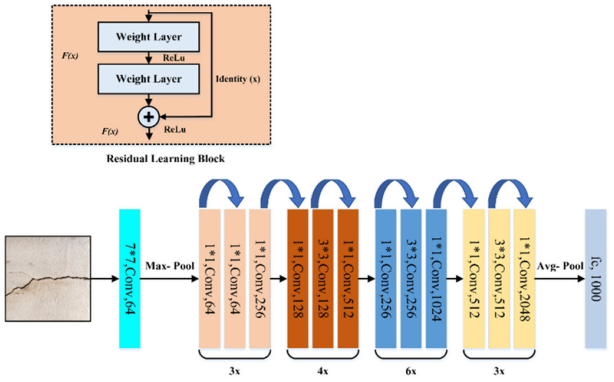

Figure 2: ResNet-50 model architecture. Image taken from [23]

ResNet [7] emerged from inquiring whether stacking more layers would enable the model to learn more effectively. To address the problem of vanishing/exploding gradients, the ResNet model introduced Residual Blocks. Using skip connections, the model connects activations from one layer to subsequent layers, forming residual blocks. These blocks are then stacked together, resulting in a powerful image classification model that can be trained on large datasets and achieve state-of-the-art results.

On the other hand, the Transformer model has emerged as a deep neural network initially applied in natural language processing (NLP) tasks, primarily based on the self-attention mechanism. Due to the Transformer's success in NLP, there has been increasing interest in applying it to computer vision tasks such as classification. This led to developing the Vision Transformer model (ViT) [22]. Recent studies, such as [22, 1], have demonstrated that ViT achieves excellent results compared to state-of-the-art convolutional networks while requiring substantially fewer computational resources for training.

3 METHODOLOGY

Due to the State-Of-The-Art (SOTA) performance of pre-trained ViT models compared to CNN models in image classification tasks, we aim to take advantage of the Vision Transformers' attention-based mechanism over small datasets and noisy depth images for safe landing zone detection. Therefore, in this section, we describe the architecture of the ResNet and ViT models and the dataset we used to fine-tune both models.

3.1 Deep Residual Network (ResNet)

We implemented the ResNet-50 model pre-trained on $ImageNet1K_V2$, see Fig. 2. The ResNet-50 architecture consists of the following main components: **Input Layer:** This layer receives an image of size 224x224x3. It includes

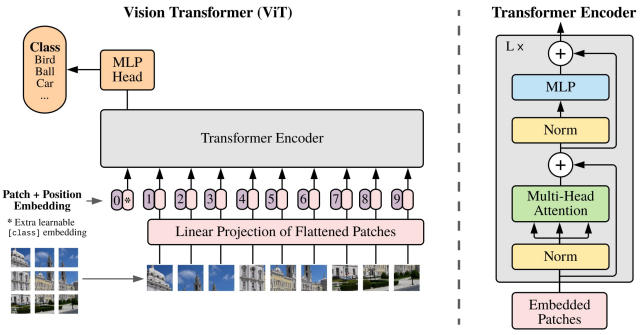

Figure 3: ViT architecture. Taken from [22]

Figure 4: Bebop Parrot 2 with drone equipment specifications.

a 7x7 convolutional layer with a stride of 2, followed by batch normalisation and a ReLU activation function. This layer extracts basic features from the input image and is followed by a 3x3 max-pooling layer, also with a stride of 2, which downsamples the output. **Residual Blocks:** Include bottleneck blocks, each with three convolutional layers (1x1, 3x3, 1x1). The 1x1 convolutions reduce and restore the dimensions, while the 3x3 convolution processes the spatial features. Each convolutional layer is followed by batch normalisation and a ReLU activation function. Shortcut connections add the input of the block directly to the output, allowing the network to learn residual functions and mitigating the vanishing gradient problem. **Fully Connected Layer:** This layer maps the output to the desired number of classes using a softmax activation function.

3.2 *Vision Transformer*

The Vision Transfomer (ViT) model takes the height and width dimensions of the input image as parameters (H and W, respectively) and works with a specified number of channels (C). The ViT transformer divides the input image into N patches with a resolution of (P, P), where the number of patches is defined as:

$$N = HW/P^2 \tag{1}$$

This N value serves as the effective input sequence length for the Transformer. The ViT model maintains a constant latent vector size D, where patches are flattened and mapped to D dimensions. Each image patch is tokenized and processed by a Transformer encoder, which represents the image. The attention head is implemented by a multi-layer perceptron (MLP) with one hidden layer. Position embeddings are added to the patch images to retain positional information. The transformer encoder consists of alternating multi-head attention layers and MLP blocks, see Fig. 3.

3.3 *Extended Synthetic and Photogrammetric Aerial-Image Dataset (ESPADA)*

We used the ESPADA dataset [5] to train ResNet and ViT classification models. ESPADA is an aerial image dataset to train deep neural networks for depth image estimation from a single aerial image.

ESPADA contains image pairs comprising chromatic images and their corresponding depth images, created from synthetic scenes and photogrammetric models imported into the AirSim simulator [24]. The camera view is top-down, and the image resolution for both RGB and Depth images is 640x480 pixels, with a field of view of 80°. Image overlap ranges between 80 and 90 per cent, depending on the drone's height. The dataset includes 80,000 RGB-D image pairs across 49 scenes: 35 from photogrammetric models and 14 from synthetic scenes, divided into five categories: urban, neighbourhood, rural, agriculture, and field. ESPADA is pre-split into training and evaluation datasets. Additionally, it contains three aerial video sequences captured with a drone flying over real scenes, acquired from heights ranging from 30 to 120 metres. For our purposes, we used solely images generated with photogrammetric models as it is argued by the authors that these images provide a more photorealistic and closer resemblance to real aerial images [5].

3.4 *Oak-D dataset*

Based on [25] framework, we used the Parrot Bebop 2 drone equipped with an Oak-D sensor, as the drone's camera is monocular, and an Intel Compute Stick as the host, depicted in Fig. 4. Additionally, the Bebop drone cannot capture top-down views as required due to the absence of a gimbal. To address this limitation, we positioned the Oak-D sensor at the front of the vehicle, facing downwards, to capture top-down views. We reduced the weight of the Oak-D sensor by replacing the camera case with a 3D-printed piece to avoid instabilities due to the Oak-D camera position, as it was done as well in [25].

The depth camera has a minimum range of 0.40 metres and a maximum range of approximately 8 metres. The Oak-D dataset contains 750 paired depth and monocular images at a resolution of 640x480 pixels collected from four flight sequences captured at heights between 0.50 metres and 2 metres. These sequences are divided into four categories: Desk,

Figure 5: Examples of aerial images and their corresponding estimated depth used for training. Note that planar surfaces exhibit a similar red tone, ideal for landing. In the depth images, the farther the depth for the camera, the lighter the red colour.

Box, Office, and Hall. The first two datasets consist of sequences flying over obstacles corresponding to their names, with 131 and 55 images, respectively. The last two datasets comprise sequences recorded in an Office (369 images) and a Hall (195 images).

4 EXPERIMENTAL FRAMEWORK

In this section, we describe the evaluation results using two different datasets: ESPADA and our small depth image dataset captured with an Oak-D sensor mounted on the drone as described in Section 3.

For each dataset, we conducted two evaluations: one with fewer images and another with a larger number. No data augmentation techniques were applied. This approach was taken to assess the impact of training the neural models with limited data, and with noisy data in the case of the images captured with the OAK-D sensor. This is particularly relevant in scenarios where there is insufficient time to collect extensive data, yet it is essential to train a model that can effectively perform tasks such as landing zone detection, even with the limited data available.

For the two experiments conducted with the ESPADA dataset, we selected sets of 200 and 400 depth images evenly split between landing and no-landing zones. Examples of these images are shown in Fig. 5. The model was trained using an 80-20% split for training and testing, respectively; we show two accuracy columns in all the tables: 'Training accuracy' was calculated immediately using a 20% split of the images for evaluating, while the other accuracy column, 'Test accuracy', was obtained by comparing the model's predictions against 150 unseen labelled images. It is important to note that these 150 unseen images were selected from the ES-

PADA dataset, specifically from the validation image subsets from the five categories in ESPADA, all captured at a height of 30 metres.

Furthermore, we conducted two additional experiments. This time, the ResNet and ViT models were trained with two small datasets taken from the Oak-D dataset, with 84 and 284 images. Examples of these images are shown in Fig. 6. As before, we selected 80% for training and 20% for testing. We evaluated the model against 120 unseen labelled images for 'Test accuracy'. It's noteworthy that the images from the Oak-D camera were noisier compared to those from the ESPADA dataset.

Experiments conducted with the ESPADA dataset, summarised in Table. 1 and 2, reveal the following observations. For the ResNet model, due to the limited information in the 200-image dataset, the model achieves its highest accuracy within the first 20 epochs. However, as the number of epochs increases, accuracy decreases, likely because the model falls into a local minimum. This behaviour was consistently observed in most of the training runs for the ResNet model. On the other hand, the ViT model, with the same 200-image dataset, reaches higher accuracy in 20 epochs and still increases its accuracy.

Experiments conducted with the 400-image dataset showed better performance for the ResNet model compared to the when using a smaller dataset, as ResNet struggles with limited information. However, the ViT model still achieved superior results when evaluated using real images.

To evaluate the performance of both models in environments different from those represented in ESPADA, we captured a dataset in an indoor scenario using the OAK-D camera onboard a MAV, as described in Section 3. In Table 3, the first row shows the results of training the ResNet model with the Oak-D small dataset of 84 images and its corresponding accuracy. We conducted this training multiple times and selected the model with the best training accuracy at 50 epochs. This selection was made because increasing the number of epochs led to a decrease in accuracy or caused the accuracy to oscillate. For the second row, with more images for training, we can observe the ResNet model performance is better.

Similar to the ResNet experiment, in Table 4, the first row shows the results of training the ViT model with the Oak-D small dataset of 84 images and its corresponding accuracy. Both training and test accuracies are reported at 50 epochs. The ViT model demonstrates faster convergence during training and performs better than the ResNet model trained with the same small number of examples. The ViT model can reach 80% accuracy within just 15 to 20 epochs.

Finally, in terms of processing time, the ResNet model runs at 8.3 ms, while the ViT model runs at 11.9 ms on average, both on outdated GPU hardware (GeForce GTX 1070) using CUDA 12.2. This is encouraging for future work, as we aim to run these models on low-budget processors. We note that, due to library issues and time constraints, we were unable to import the models to run on the OAK-D camera, which has a GPU. However, we plan to pursue this effort to fully utilise the sensor's capabilities.

ResNet Trained on ESPADA			
Images	**Epochs**	**Training Acc**	**Test Acc(%)**
200	20	51.28	**57.27**
	30	53.85	32.7
	40	62.12	30.5
	50	70.9	30
400	20	68.18	67.27
	30	70.9	70.0
	40	84.62	84.54
	50	87.18	**86.9**

Table 1: Accuracy of the fine-tuned ResNet-50 model trained with ESPADA. Note that the model requires more images and more epochs to achieve more accuracy, highlighted in bold. A total of 100 images were used as test set.

ViT Trained on ESPADA			
Images	**Epochs**	**Training Acc**	**Test Acc(%)**
200	20	86.67	81.9
	30	85.85	83.63
	40	89.12	86.09
	50	91.9	**87.0**
400	20	80.9	79.8
	30	86.36.0	82.36
	40	96 .36	90.9
	50	96.45	**91.3**

Table 2: Accuracy of the ViT model trained with ESPADA. Note that the model achieves superior results with fewer images compared to ResNet, and the best results when using more images for training, highlighted in bold.

ResNet Trained on captured dataset			
Images	**Epochs**	**Training Acc**	**Test Acc(%)**
84	50	70.9	62.4
284	50	94.9	**96.6**

Table 3: Accuracy for the ResNet model trained with images captured with the OAK-D camera onboard the MAV.

5 CONCLUSION

We have presented a methodology to address the problem of landing zone detection for MAVs. To achieve this, we have explored the use of two popular Deep Neural Networks,

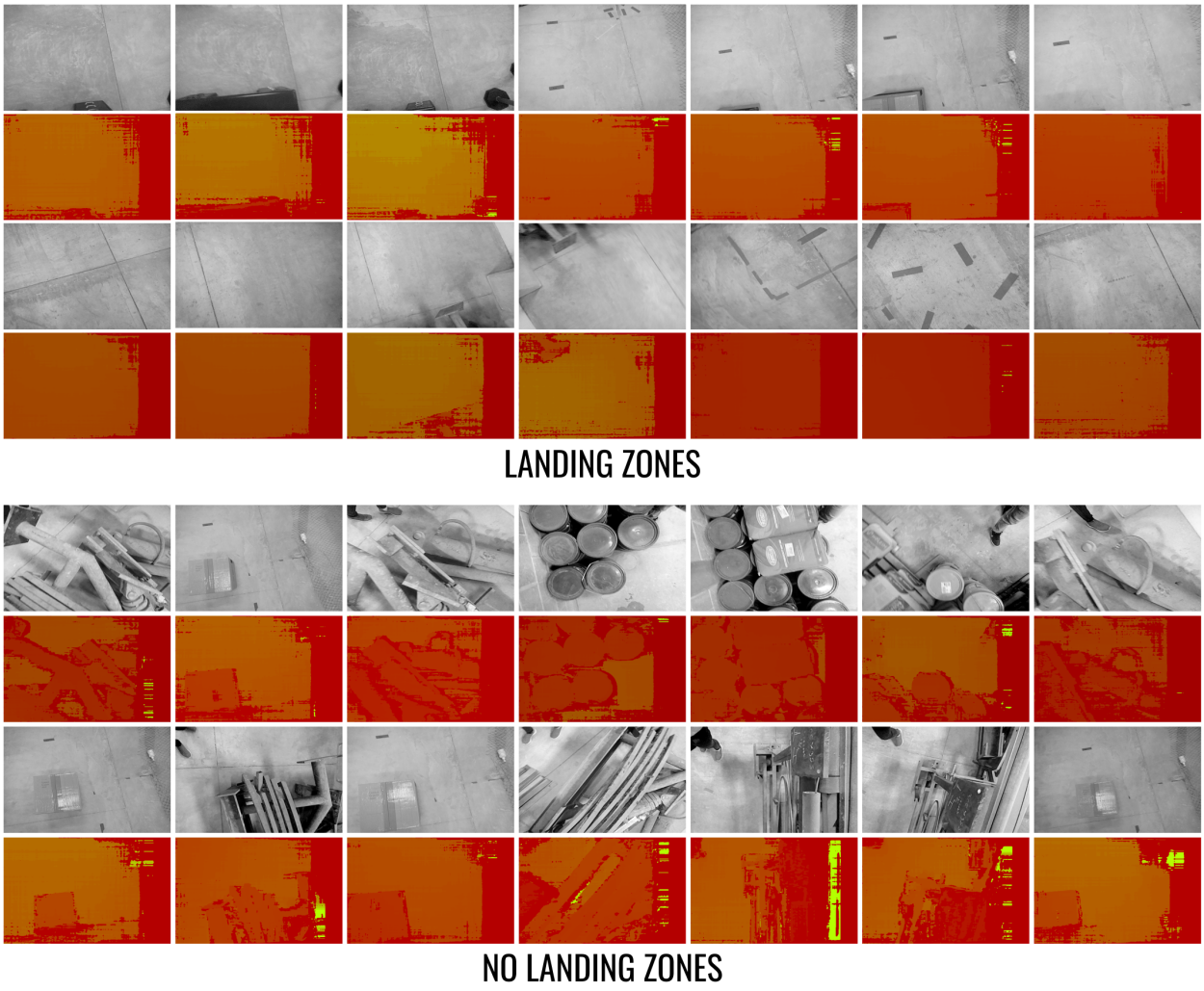

Figure 6: Examples of aerial images taken from our dataset and their corresponding estimated depth used for training. Note that planar surfaces, ideal for landing, exhibit a similar red tone, and Oak-D images are noisier than ESPADA dataset. In the depth images, the farther the depth for the camera, the lighter the red colour. Taken from [5]

ViT Trained on captured dataset			
Images	**Epochs**	**Training Acc**	**Test Acc(%)**
84	50	82.7	73.63
284	50	98.2	**96.36**

Table 4: Accuracy of the ViT model trained with images captured by the OAK-D camera onboard the MAV. Note that this model achieved better results than ResNet with fewer images.

namely Residual Networks and Vision Transformers. The former is a Convolutional Neural Network that has proven highly successful in various computer vision tasks. However, the latter offers an architecture capable of capturing spatial relations among visual data, thanks to its attention mechanism. Therefore, for computer vision tasks, it may offer superior performance when trained with fewer examples.

These findings were confirmed in our experiments, where we trained both models using aerial images from a state-of-the-art dataset known as ESPADA and images captured with the OAK-D sensor, a depth camera mounted on a MAV. We opted to use depth images in both cases, as the literature suggests that depth images provide richer data in the form of depth values, aiding in determining whether the observed scene corresponds to a landing zone [6].

Although CNNs have a strong track record in various computer vision tasks and are efficient with large-scale

datasets, we have also explored the use of both models trained from scratch and pre-trained models. The choice to employ Vision Transformers (ViTs) and to conduct a more in-depth analysis is grounded in previous studies that contrast ViTs and CNNs [26, 27, 28]. Based on these works and our experiments, our results indicate that pre-trained Vision Transformers excel in scenarios where understanding global dependencies and context is crucial. Vision Transformers typically require larger amounts of training data to achieve comparable performance to CNNs. Our findings reinforce that the ViT model surpasses the ResNet model, particularly when pre-trained, resulting in better accuracy when fine-tuned with fewer images. It is also noteworthy that the depth images from the OAK-D camera were less accurate than those from other depth cameras, such as Intel's RealSense. Yet, the ViT model effectively managed the noisy data. Moreover, with an average prediction time of $11.9ms$ on an outdated GPU, the ViT model shows potential for real-time detection when implemented on state-of-the-art processors aboard the vehicle.

Future work will involve testing depth images generated by a Deep Neural Network from a single image, as outlined in [29]. Additionally, we will explore distillation strategies for the ViT to enhance prediction speed on embedded hardware. Furthermore, we will investigate Explainable Artificial Intelligence methods to provide a more in-depth analysis of feature extraction characteristics in both ViT and CNN models.

ACKNOWLEDGEMENTS

The first author is thankful to the National Council of Science and Technology of Mexico (CONAHCYT) for her scholarship number 1026123.

REFERENCES

[1] Y. et al. Liu. A survey of visual transformers. *IEEE Transactions on Neural Networks and Learning Systems*, 35(6):7478–7498, June 2024.

[2] Leticia Oyuki Rojas-Perez and Jose Martinez-Carranza. Towards autonomous drone racing without gpu using an oak-d smart camera. *Sensors*, 21(22):7436, 2021.

[3] L Oyuki Rojas-Perez and Jose Martinez-Carranza. Deeppilot4pose: a fast pose localisation for mav indoor flight using the oak-d camera. *Journal of Real-Time Image Processing*, 20(1):8, 2023.

[4] M. Hermann, B. Ruf, M. Weinmann, and S. Hinz. Self-supervised learning for monocular depth estimation from aerial imagery. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, V-2-2020:357–364, 2020.

[5] Rafael Lopez-Campos and Jose Martinez-Carranza. Espada: Extended synthetic and photogrammetric aerial-image dataset. *IEEE Robotics and Automation Letters*, 6(4):7981–7988, 2021.

[6] L Oyuki Rojas-Perez, Roberto Munguia-Silva, and Jose Martinez-Carranza. Real-time landing zone detection for uavs using single aerial images. In *10th international micro air vehicle competition and conference, Melbourne, Australia*, pages 243–248, 2018.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[8] J. Martinez-Carranza, E. Inzunza-Gonzalez, E. E. Garcia-Guerrero, and E. Tlelo-Cuautle. *Machine Learning for Complex and Unmanned Systems*. CRC Press, 2024.

[9] M. S. Alam and J. Oluoch. A survey of safe landing zone detection techniques for autonomous unmanned aerial vehicles (uavs). *Expert Systems with Applications*, 179:115091, 2021.

[10] J. Lim, M. Kim, H. Yoo, and J. Lee. Autonomous multirotor uav search and landing on safe spots based on combined semantic and depth information from an onboard camera and lidar. *IEEE/ASME Transactions on Mechatronics*, 2024.

[11] F. Neves, L. Branco, M. Pereira, R. Claro, and A. Pinto. A multimodal learning-based approach for autonomous landing of uav. *arXiv preprint arXiv:2405.12681*, 2024.

[12] J. Chen, W. Du, J. Lin, U. M. Borhan, Y. Lin, B. Du, and J. Li. Emergency uav landing on unknown field using depth-enhanced graph structure. *IEEE Transactions on Automation Science and Engineering*, 2024.

[13] 2014 International Conference et al. Vision-based autonomous landing system for unmanned aerial vehicle: A survey. In *2014 International Conference on Multisensor Fusion and Information Integration for Intelligent Systems (MFI)*, pages 1–8. IEEE, 2014.

[14] Aldrich A Cabrera-Ponce and Jose Martinez-Carranza. A vision-based approach for autonomous landing. In *2017 Workshop on Research, Education and Development of Unmanned Aerial Systems (RED-UAS)*, pages 126–131. IEEE, 2017.

[15] C. Theodore, D. Rowley, A. Ansar, L. Matthies, S. Goldberg, and D. Hubbard. Flight trials of a rotorcraft unmanned aerial vehicle landing autonomously at unprepared sites. In *Annual Forum Proceedings-American Helicopter Society*, page 1250. AMERICAN HELICOPTER SOCIETY, INC, 2006.

[16] M. Meingast, C. Geyer, and S. Sastry. Vision based terrain recovery for landing unmanned aerial vehicles. In *2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No. 04CH37601)*, pages 1670–1675. IEEE, 2004.

[17] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: part i. *IEEE Robotics & Automation Magazine*, 13(2):99–110, 2006.

[18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[19] Yann LeCun, Bernhard Boser, John S. Denker, Donnie Henderson, Richard E. Howard, Wayne Hubbard, and Lawrence D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1989.

[20] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional neural networks. In *European Conference on Computer Vision (ECCV)*, 2014.

[21] YuYu Zheng, HaoXuan Huang, and Junming Chen. Comparative analysis of various models for image classification on cifar-100 dataset. *Journal of Physics: Conference Series*, 2711:012015, 2024.

[22] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, ..., and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[23] Luqman Ali, Fady Alnajjar, Hamad Jassmi, Munkhjargal Gochoo, Wasif Khan, and Mohamed Serhani. Performance evaluation of deep cnn-based crack detection and localization techniques for concrete structures. *Sensors*, 21:1688, 2021.

[24] S. Shah, D. Dey, C. Lovett, and A. Kapoor. Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In *Field and Service Robotics: Results of the 11th International Conference*, pages 621–635. Springer International Publishing, 2018.

[25] L. O. Rojas-Perez and J. Martinez-Carranza. Towards autonomous drone racing without gpu using an oak-d smart camera. *Sensors*, 21(22):7436, 2021.

[26] C. Matsoukas, J. F. Haslum, M. Söderberg, and K. Smith. Is it time to replace cnns with transformers for medical images? *arXiv preprint arXiv:2108.09038*, 2021.

[27] Y. Bai, J. Mei, A. L. Yuille, and C. Xie. Are transformers more robust than cnns? *Advances in Neural Information Processing Systems*, 34:26831–26843, 2021.

[28] Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. Transformers in vision: A survey. *ACM Computing Surveys*, 54(10s):200, January 2022.

[29] Luis Pellegrin and José Martinez-Carranza. Towards depth estimation in a single aerial image. *International journal of remote sensing*, 41(5):1970–1985, 2020.