Detecting Aircraft From Satellite Imagery

Pitt Hu
University of Bristol
Email: kd19082@bristol.ac.uk

Yuanzhuo Hu
University of Bristol
Email: fg20441@bristol.ac.uk

Aaron Vinod
University of Bristol
Email: ce20480@bristol.ac.uk

Adam Morris
University of Bristol
Email: ks20447@bristol.ac.uk

Georgia Hadjidemosthenous
University of Bristol
Email: nb19638@bristol.ac.uk

***Abstract*—In recent years, object detection in satellite imagery has become essential in remote sensing applications. This study presents the development and comparison of three distinct artificial intelligent systems and their ability to correctly classify images of aircraft from real-world aerial photography. These include a customised CNN, application of the pre-trained YOLO v7 model, and a k-NN predictor using principal components analysis. The models were evaluated and compared through accuracy and F1 scores. It was found that the CNN method performed the best, closely followed by YOLO v7, with k-NN performing poorly.**

I. Introduction

Satellite imagery is a category of aerial photography which captures elements of the Earth's surface, being a key source of information for numerous different sectors of society [1] [2]. Processing of these images has remained the centre of research in the sectors of disaster management, oceanography, weather forecasting and more [3]. High-resolution images are particularly useful in detecting possible threats, in both military and emergency scenarios [4]. Although humans possess the ability to complete these tasks, it is within interest to develop techniques to automate the process with high degrees of accuracy.

Image processing remains an ongoing challenge for artificial intelligent systems. Humans can perceive and process images almost instantaneously with high accuracy, limited only by their ability to see. AI, however, requires complex algorithms to imitate this natural ability, often requiring vast computational power. This study focuses on tackling this challenge through aircraft detection in satellite imagery by classifying areas of images depending on whether an aircraft is present. The efficient and precise detection of an aircraft remains a demanding task due to the highly complex nature of image data. This complexity arises from objects occurring in different orientations, sizes, colours and quality. In addition, object orientation changes its aspect ratio, which further complicates object localisation [4].

A series of deep and supervised learning methods are presented, which aim to classify sets of images into two categories: plane and no-plane. Achieving sufficient accuracy in this binary classification problem presents many benefits, in particular with aircraft detection for both safety and recovery purposes. Future development will allow for further satellite image classifiers to be combined and used together for in-depth analysis of more general aerial photographs.

II. Literature Review

Accurate and rapid detection of aircraft, or other potential targets, remains a challenging problem despite the availability of many computer vision methods in literature [4]. The need to manually extract features from objects limits the capabilities of object detection methods. However, deep neural network-based techniques extract features automatically and are thus the preferred procedure to follow for such tasks.

Deep learning methods have shown great success in solving problems like the one set out in this study. An example is seen in the works by Mark Pritt and Gary Chern, where deep learning algorithms were applied to solve the problem of object recognition in multi-spectral satellite imagery [5]. Pritt and Chern developed a deep learning system for classifying objects from the IARPA Functional map of the World dataset into 63 different classes.

Traditional methods produce a low precision rate of aircraft image recognition when classifying these images. This is due to the existence of different types of aircraft and significant similarities between different models. To solve this problem Wang et al. propose a hybrid attention network model to implement an aircraft recognition algorithm [6].

CNN's have shown great success in other areas of image classifiers, especially within the medical sector. In the paper by Qing Li et al, convolutional neural networks were utilised to identify patches within lung screening images [7]. Similarly, opportunities for diagnosis of skin cancer through deep-learning methods were seen to be "making encouraging progress" [8].

Another recent, very popular and successful method of AI object detection comes from the You Only Look Once (YOLOv7) algorithm. The authors claim "YOLOv7 surpasses all known object detectors in both speed and accuracy" [9]. This system presents a pre-trained model on a large dataset, containing dozens of individual labels. The model can be used to identify common objects in most photographs. Furthermore, it may utilised for more specific tasks by refining its training on customised datasets.

Use of supervised learning methods for image classification has also been noticed in literature. Zhuang et al. [10] demon-

strate the development of a novel k-NN system which unifies the feature extraction process with the classification process. This method assigns each training image to the same class as its k nearest neighbours during learning feature extractor.

III. Methods

Before model training can begin, a suitable dataset is required. The data used for this project is provided through Kaggle, named the "Planes in Satellite Imagery" dataset [11]. The set contains 32,000 RGB images of size 20x20, 8000 of which are labelled 1 (plane) and the reamining 24,000 labelled 0 (no-plane).

After downloading the data, the relevant features and labels can be extracted to be used for training, validation and testing. The dataset contains the list of all 32000 PNG images, as well as 4 larger "scenes" containing examples of the original satellite images. Additonally, a JSON version file is provided of the metadata of all the contained images. This includes the RGB values, label classifier and co-ordinates, with respect to larger scenes, of each data entry.

RGB matrices are stored as a series of 1200 integers, representing a single image. This list is further split into 3 sequential sections, of length 400, corresponding to the red, green and blue components. The data is therefore required to be reshaped into an appropriate image format. Each component is converted to a 20x20 matrix and concatenated together. The size of the final dataset is 32000 x (20x20x3) features, each with a corresponding label.

The chosen training, validation and test dataset length split is 80:10:10 (%) respectively. This is a typically used ratio that will allow for sufficient training for the created models. In addition, a validation set will allow for hyperparameter tuning outside of training and test accuracy. Altercations made after testing will not reflect the models practical accuracy when given further unseen data. This dataset will therefore only be used for final model evaluations.

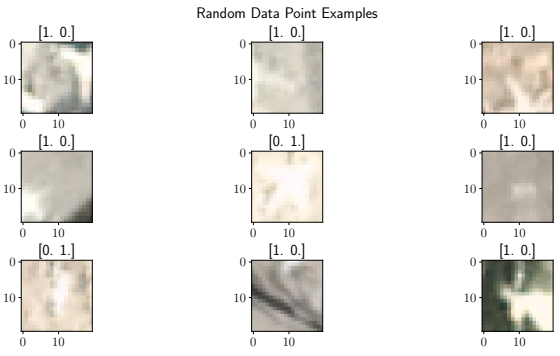

Fig. 1. Random training set examples. [0.1.] indicates a plane

Figure 1 shows nine random samples that have been visualised from the training set. It is clear that resolutions of this scale make it difficult for humans to identify which images contain planes. Additionally, unsupervised methods may struggle to categorise individual images due to lack of distinguishability between them. Three different deep-learning and supervised methods have therefore been considered for this project, convolution neural network, YOLO v7 and k-NN with principal component analysis. Initial predictions are that the CNN and YOLO models will perform the best, and k-NN would struggle with this task.

A. Convolutional Neural Network (CNN)

A CNN's built in convolutional layer decreases the dimensionality of input data without overlooking its relevant information [12]. CNNs as a result are very successful in minimising the number of parameters while the quality of models is maintained [13]. This is especially useful within images, due to the large number of parameters associated with their features, being the RGB values.

The developed CNN uses 8 sequentially stacked layers of four different types. These include 2D convolutional, 2D maximum pooling, flatten and dense layers. Each layer uses the rectified linear unit activation function, except for the final which uses softmax. The set kernel size for convolutional layers is (5, 5), and a pool size of (2, 2) for pooling layers. Finally, the model uses the "Adam" stochastic gradient descent optimisation method on compiling. The model is trained on the RGB matrices dataset and the assigned label vector for these images. This gives an input shape of height x width x depth.

Model training and validation occurs at the end of each epoch, where the model updates neuron weightings based on the results of the iteration. More epochs generally increase model accuracy at the expense of time. However, too many can also lead to over-fitting of the training data. It is important therefore to choose an appropriate number that achieves sufficient accuracy in the shortest time. The results of training for the constructed CNN have been plotted as a function of the number of epochs.

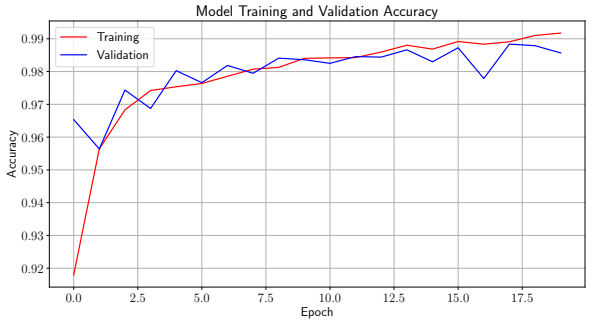

Fig. 2. Model training and validation accuracy over 20 epochs

Fig. 3. Model training and validation binary cross-entropy loss over 20 epochs

Figure 2 illustrates the accuracy achieved by the model at the end of each epoch on both the training and validation set. Accuracy gives a metric of how many mistakes the model makes against the ground truth. The model initially produces an accuracy of around 91.8% and 96.5% respectively, with a maximum accuracy achieved of 99.2% and 98.7%. Binary cross-entropy loss is plotted in Figure 3, as only two labels are involved in classification. This gives an indication as to how far from the true value the model predictions are. It is therefore within interest to minimise both of these functions. Although validation accuracy generally increases with higher epochs, loss begins to fluctuate and increase, indicating the model is beginning to over-fit the training data. It is therefore not suited to use 20, or more, epochs. Average epoch training time on the used hardware is around 25 seconds, further deterring the use of higher iterations. It was therefore determined that 12 epochs produces sufficient accuracy and loss in a practical time frame, which the final model has been trained using.

B. You Only Look Once-v7 (YOLO-v7)

The YOLO v7 model is an open-source artificial intelligence system for object detection and labelling. The algorithm splits images into smaller cells to create bounding boxes and their predicted class probabilities. Named the "trainable bag-of-freebies", the involved methods aim to optimise the structure and training process of previously designed object detection programs without increasing inference cost [14]. YOLO was chosen for this project due to its ease of implementation and utilisation of a pre-trained, highly accurate system.

Although the model has been previously trained on a large dataset, YOLO can be fine-tuned on custom data to further develop its capabilities on unique tasks. However, data must be passed to the model in a specific TXT file format, which is not present in this project's dataset. To overcome this issue, Roboflow has been used to create the required training, validation and test sets. Roboflow allows users to upload and manually annotate images to be used for AI object detection. Additonally, pre-processing and augmentation techniques can be included to expand training data.

The four included satellite image scenes were used as the datasets for this section of the project. These images illustrate a large area containing many stationary, grounded aircraft of different shapes, sizes and orientations. Each image was uploaded to Roboflow, where bounding boxes for each plane were drawn and labelled accordingly. Pre-processing resized all images to a consistent size of 640x640 pixels, being YOLO's default training size. Image augmentation was also applied to the training set, including saturation, brightness and rotation of images to generate further examples. The produced datasets were split into the same train/validation/test proportions as for the CNN and passed to the YOLO v7 model.

Model training lasted for 25 epochs, utilising Google Colab's GPU processing, with a total training time of 0.178 hours, similar to that of the the CNN model. Final training produced an average mean precision of 96%, and a validation precision of 95.9%. An example output can be seen in figure 4, showing how the model produces bounding boxes with the associated confidence interval. This confidence interval relates to the probability that the model predicts a certain class is present within the bounding box. Increasing the size of the bounding box generally increases this confidence at the expense of positional accuracy.

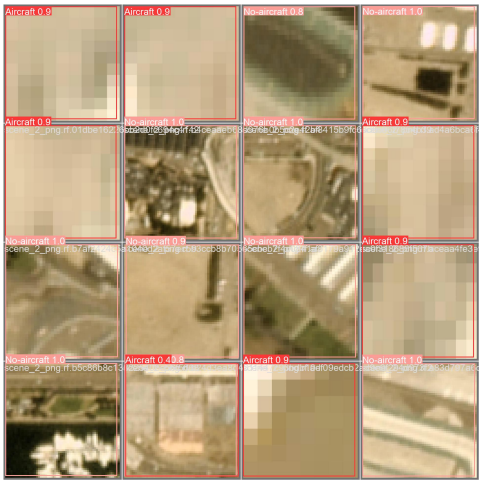

Fig. 4. Example output of YOLOv7 Model

C. k-NN Classification with Reduced Features dataset

Principal Component Analysis (PCA) was used to reduce the dimensionality of the dataset since it is among the most favoured algorithms for linear dimension reduction. In machine learning dimensionality indicates the number of features in the given dataset [15]. Dimensionality reduction prevents the model from overfitting in addition to reducing its complexity [16]. After using PCA, the 1200 pixels representing the features are reduced to 18, which still capture around 90% of the variance of the data. This dataset of reduced features is then classified with the k-Nearest Neighbours (k-NN) algorithm. Its training process involves storing feature vectors and labels of the training images. The algorithm assigns the class that is the most common in the k nearest neighbors to the unlabeled data point [17].

To determine the value of k, a commonly used heuristic know as the elbow method is used. This is used as a cutoff

point for the number of clusters, where diminishing returns provide no further value with added cost. This is a simple technique that aims to avoid overfitting to the training set.

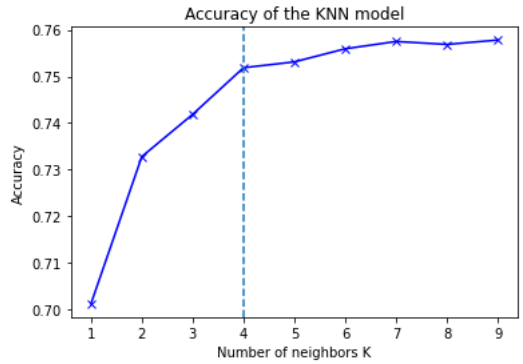

Fig. 5. k-NN accuracy on the validation set as a function of k. Dotted line indicates the 'elbow' point of the graph.

Figure 5 shows an accuracy plot of the k-NN model as a function of the number of neighbours. Using the principle of the elbow method, k=4 was chosen as the most suitable value. Using this, the k-NN model produces an accuracy on the training and validation dataset of around 95.9% and 75.2% respectively.

IV. Testing and Analysis

To compare our methods 4 different metrics were considered: Accuracy, the proportion of the samples that were predicted correctly; Recall, the true positive rate; Precision, the number of positive class predictions that truly belong to the positive class; F-1 score, the harmonic mean of the precision and recall. The F1 score was determined to be the best metric for the dataset, since it factors in the distribution of the class samples, which is unaccounted for in measures such as accuracy. Because it's a harmonic mean of the other 2 metrics, it punishes extreme values of either metric. These reasons allow F1 scores to be a confident performance metric for the given models. The F1 score can be calculated using equation (1).

$$F_1 = 2 \times \frac{precision \times recall}{precision + recall} = \frac{TP}{TP + \frac{1}{2}(FP + FN)} \quad (1)$$

TABLE I
F1-score for each method on the test sets

Method	CNN	YOLO v7	k-NN
F1-score (%)	93.9	90.0	26.7

As shown in the table I, the CNN and YOLO method both produce high F1 scores on the test data (93.9%, 90.0%), where as the k-NN F1 score is very low at 26.7%. As a result, further analysis of the CNN and YOLO models is required. k-NN is deemed unsuitable for this task, and therefore requires no further investigation into its use without fundamental reconstruction.

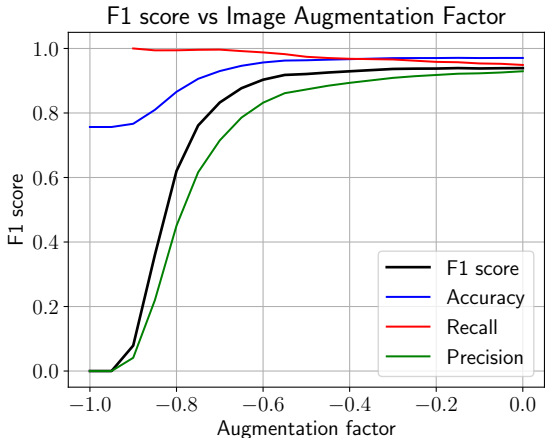

Fig. 6. F1 score, accuracy, recall and precision of CNN model against test set as a function of augmentation factor.

Figure 6, shows the CNN's performance on varying datasets, using different perturbations of the RGB values for each image (for example changing the brightness). The augmentation equation $x = x + \alpha x$ is used to change the RGB values for each image, with α being the augmentation factor and x being the image vector. As a result, α=-1 will produce an entirely black square, and α=0 is the original image. Examples of this can be seen in figure 7.

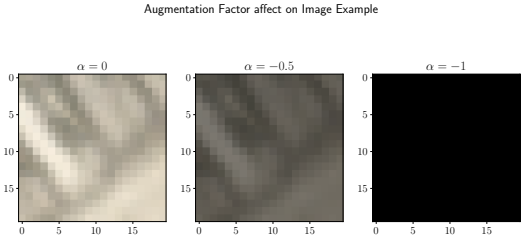

Fig. 7. Visualisation of Augmentation factor affect on an example image

This graph enforces how the F1 score is a better indicator of performance then other metrics. Accuracy remains high even when the precision is very low, and precision is low when recall is high. F1 captures all elements of this, giving a more representative result. The CNN performed well up to an augmentation factor of around -0.6, maintaining a high F1 score. Below this value the CNN's abilities diminishes quickly. This, however, cements the capabilities of the CNN to classify a large array of aircraft in many different scenarios. CNN is determined to be a successful and practical technique for this project task.

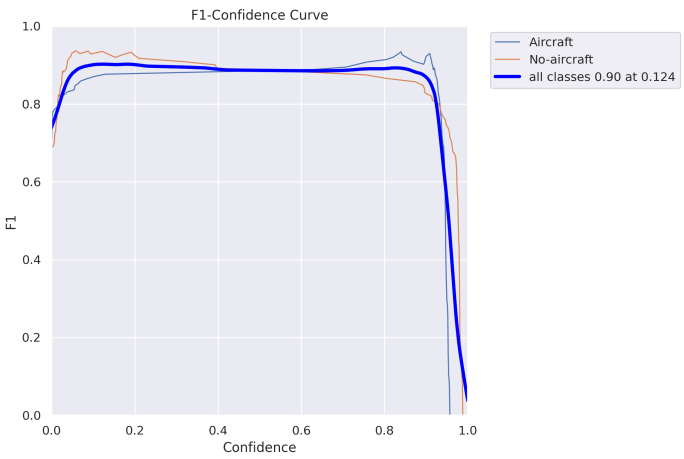

Fig. 8. YOLO v7 F1-confidence plot on test set

In Figure 8, the F1-confidence plot on the test data provides information on the YOLO model's overall performance. The plot shows the relationship between the F1 score and the confidence threshold used for object detection. The plot indicates a high F1 score of 0.90 at low confidence thresholds, maintaining approximately the same value up to around 0.88. This indicating that the model can detect objects accurately with high certainty and positional accuracy. Overall, the model is determined to work well and does not require further optimisation.

The k-NN model's F1 score was calculated to be 26.7%, as seen in table I. The accuracy in training and validation seemed promising, however this arises from the uneven split of labels, meaning high accuracy can be achieved by only predicting no-plane. As a result, this model choice is not suggested for this task.

V. Discussion

Section IV provides the measures on how to compare the effectiveness and practicality of the different methods. It was found that the convolutional neural network model performed the best at this task, with an F1 score of 93.9% on the test dataset. YOLO closely followed with an average F1 score of 90% across a large range of confidence intervals. k-NN performed significantly worse with an F1 score of 26.7%.

To understand how the CNN is able to produce the highest F1 score, figure 6 shows how the test dataset was modified with an augmentation factor. The aim for this manipulation is to understand what factors influenced the CNN's classification ability. Analysis showed that the CNN was able to maintain performance for a range of augmentation values. This suggests that the CNN is able to recognise the appropriate patterns even as image clarity reduces. Extreme changes hinders the CNN's ability as the images become indistinguishable, as illustrated in figure **??**. At smaller magnitudes of augmentation values, the variance of the pixels is maintained as the changes to the RGB values are fairly consistent. However, at the extreme end, values tend to 0, lowering the image variance, and hence any recognisable patterns. Initial hypothesis for this manipulation was the affect on performance would be minimal, up until a critical point when images become discernible. This was predicted due to the dataset encompassing a diverse range of images that naturally encapsulate various factors, such as rotation, aspect ratio, sizes, shapes, and cut-offs. This may also be achieved manually by performing such adjustments to training examples, as was seen in the use of Roboflow. However, the quality of the provided data meant this was unnecessary for the CNN. Feature-rich datasets allow for the development of robust object detection models that can effectively detect and classify aircraft in real-world aerial photography. This was deemed as a key factor contributing to the excellent performance of the CNN and YOLO models.

CNN and YOLO had similar training times on their respective datasets. However, taking into account the need to re-create and modify the original dataset for YOLO v7, this presented a more limiting use in the models practicality. On the other hand, CNN cannot perform on images not of the same size as the original dataset, meaning it lacks robustness when compared to YOLO. YOLO also provides a better user interface, generating bounding boxes of the predicted objects' position.

The dataset used in this project has a high degree of usability, having sufficient data and room to develop future models by using additional features provided, such as the coordinate system. The dataset could be improved by storing the RGB values as matrices instead of long lists, making it easier to deploy on other image classifying models. Additionally, there is an uneven split of classification, having 3 times as many no-plane as opposed to plane labels. This has partially been taken into account by use of F1 score, however, an even distribution would produce more un-biased results when predicting plane classes.

These results agree with the initial predictions set out at the beginning of the project. It would therefore be advised to use a CNN model approach for this task.

VI. Conclusion

This study provides an overview of 3 artificial intelligence algorithms while assessing their performance on classifying satellite imagery and detecting aircraft. Even though the precise detection of aircraft is challenging, recently developed deep learning-based methods for object detection, such as YOLO, and CNNs have demonstrated promising results. However, the supervised learning technique implemented on the dataset, k-NN, performed significantly worse.

VII. Future Work

Based on the results of the CNN, there is very little room for improvement. However, this model may be combined with other CNN's to further categorise images within satellite photography.

Based on the YOLO model performance, the accuracy has small room for improvement, especially compared to that of the CNN. Researchers may focus on adding more

image-level augmentations. The model learns by generating augmented versions of each image in the training data set. Rotations, noise and occlusions occur when objects in an image overlap, making it difficult for the model to detect them. Future developments may improve the model's ability to handle occlusions and accurately detect objects even when they are partially obscured.

Furthermore, YOLO performs well in detecting more prominent objects, but detecting smaller targets like aircraft in satellite images can be challenging. To improve the detection of small objects, researchers may explore strategies such as adjusting the model architecture, changing the loss function, or incorporating additional training data with smaller objects. Another approach is to use object detection algorithms specifically designed for small objects, such as the Faster R-CNN or RetinaNet models. Future developments may focus on improving the model's ability to accurately detect and classify smaller objects.

A variety of other methods also exist within image classification. An upcoming technique known as Vision Transformers has shown promising results as an object detection method [18]. This could quickly become the benchmark for these types of models, providing better comparisons for traditionally used techniques.

The further application of the YOLO model and CNN in detecting aircraft from satellite images could have significant implications in several domains, such as military intelligence and aviation safety. These two models could identify aircraft and their real-time movements from satellite images and videos. This information could be used to gather intelligence to detect potential threats and accidents. Moreover, those models could detect aircraft in distress or deviating from their flight path for aviation safety agencies. This information could be used to initiate search and rescue operations or alert authorities to potential safety risks. Airports may also utilise these systems by detecting disrupting objects near aircraft. Satellite videography may also provide practical values in supporting airport flow monitoring and scheduling.

Overall, the YOLO model and CNN have the potential to revolutionise detection of aircraft from satellite images, enabling faster, more accurate, and cost-effective monitoring in various domains.

REFERENCES

[1] "11+ amazing uses of satellite imagery," August 2021. [Online]. Available: https://www.spatialpost.com/uses-of-satellite-imagery

[2] B. Enos, "Satellite imagery: Reasons to use satellite imagery services," March 2022. [Online]. Available: https://www.enostech.com/satellite-imagery-reasons-to-use-satellite-imagery-services/

[3] A. Tahir, M. Adil, and A. Ali, "Rapid detection of aircrafts in satellite imagery based on deep neural networks," *arXiv preprint arXiv:2104.11677*, 2021.

[4] B. Azam, M. J. Khan, F. A. Bhatti, A. R. M. Maud, S. F. Hussain, A. J. Hashmi, and K. Khurshid, "Aircraft detection in satellite imagery using deep learning-based object detectors," *Microprocessors and Microsystems*, vol. 94, p. 104630, 2022.

[5] M. Pritt and G. Chern, "Satellite image classification with deep learning," in *2017 IEEE applied imagery pattern recognition workshop (AIPR)*. IEEE, 2017, pp. 1–7.

[6] Y. Wang, Y. Chen, and R. Liu, "Aircraft image recognition network based on hybrid attention mechanism," *Computational Intelligence and Neuroscience*, vol. 2022, 2022.

[7] Q. Li, W. Cai, X. Wang, Y. Zhou, D. D. Feng, and M. Chen, "Medical image classification with convolutional neural network," in *2014 13th International Conference on Control Automation Robotics & Vision (ICARCV)*, 2014, pp. 844–848.

[8] M. Goyal, T. Knackstedt, S. Yan, and S. Hassanpour, "Artificial intelligence-based image classification methods for diagnosis of skin cancer: Challenges and opportunities," *Computers in Biology and Medicine*, vol. 127, p. 104065, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0010482520303966

[9] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," 2022. [Online]. Available: https://arxiv.org/abs/2207.02696

[10] J. Zhuang, J. Cai, R. Wang, J. Zhang, and W.-S. Zheng, "Deep knn for medical image classification," in *Medical Image Computing and Computer Assisted Intervention–MICCAI 2020: 23rd International Conference, Lima, Peru, October 4–8, 2020, Proceedings, Part I 23*. Springer, 2020, pp. 127–136.

[11] "Planes in satellite imagery." [Online]. Available: https://www.kaggle.com/dsv/4804225

[12] N. Lang, "Using convolutional neural network for image classification," December 2021. [Online]. Available: https://towardsdatascience.com/using-convolutional-neural-network-for-image-classification

[13] P. Mishra, "Why are convolutional neural networks good for image classification?" May 2019. [Online]. Available: https://medium.datadriveninvestor.com/why-are-convolutional-neural-networks-good-for-image-classification-146ec6e865e8

[14] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," *arXiv preprint arXiv:2207.02696*, 2022.

[15] N. S. Chauhan, "Dimensionality reduction with principal component analysis (pca)." [Online]. Available: https://www.kdnuggets.com/2020/05/dimensionality-reduction-principal-component-analysis.html

[16] L. Li, "Principal component analysis for dimensionality reduction," May 2019. [Online]. Available: https://towardsdatascience.com/principal-component-analysis-for-dimensionality-reduction-115a3d157bad

[17] I. Nurwauziyah, S. UD, I. G. B. Putra, and M. I. Firdaus, "Satellite image classification using decision tree, svm and k-nearest neighbor," *no. July*, 2018.

[18] X. Chen, C.-J. Hsieh, and B. Gong, "When vision transformers outperform resnets without pretraining or strong data augmentations," *arXiv preprint arXiv:2106.01548*, 2021.