Urban Tree Detection and Species Classification Using Aerial Imagery

Mahdi Maktab Dar Oghaz[1], Lakshmi Babu Saheer[1], and Javad Zarrin[1]

Faculty of Science and Engineering, Anglia Ruskin University, Cambridge, UK
Mahdi.maktabdar@aru.ac.uk

Abstract. Trees are essential for climate change adaptation or even mitigation to some extent. To leverage their potential, effective forest and urban tree management is required. Automated tree detection, localisation, and species classification are crucial to any forest and urban tree management plan. Over the last decade, many studies aimed at tree species classification using aerial imagery yet due to several environmental challenges results were sub-optimal. This study aims to contribute to this domain by first, generating a labelled tree species dataset using Google Maps static API to supply aerial images and *Trees In Camden* inventory to supply species information, GPS coordinates (Latitude and Longitude), and tree diameter. Furthermore, this study investigates how state-of-the-art deep Convolutional Neural Network models including VGG19, ResNet50, DenseNet121, and InceptionV3 can handle the species classification problem of the urban trees using aerial images. Experimental results show our best model, InceptionV3 achieves an average accuracy of 73.54 over 6 tree species.

Keywords: Urban Tree Detection, Convolutional Neural Network, Aerial Imagery

1 Introduction

Trees are well recognised for their importance to the climate and human life. Environmentally, trees slow surface runoff from rainfall, reducing the risk of flood, water pollution and soil erosion [4]. In urban areas, trees improve overall air quality by absorbing particulate matter and create a cooling effect which helps in adapting to the "heat island" effect [17]. Moreover, urban trees play a key role in climate change adaptation or even mitigation by reducing CO2 levels, the main contributor to climate change. Urban trees also improve the perception of an area by blocking noise, dust, wind and glare [7]. Studies indicate, urban trees can reduce indoor heating and cooling expenses by blocking the wind, weather and casting shade around the housing area [33]. In order to exploit this potential, effective forest and urban tree monitoring and management is essential. This requires information about composition, species, age, health, and location of trees which helps in better planning of plantation programs, growth monitoring and pruning. This also facilitates biodiversity of the vegetation and promotes robust ecosystem with greater resilience to disease and pests and better productivity [9, 1]. Such management system demands for a reliable yet economically viable platform to automatically detect, classify and monitor forests and urban trees to guide policy makers to devise better long term management strategy and ensures long term sustainability. Historically, experts and volunteers on the ground were in charge of this laborious and time-consuming management system. However, advent of aerial, satellite and LiDAR imagery has now put a new dimension to these practices[25]. LiDAR technology repeatedly used to estimate the number of trees in an area [32] and categorise their species [12, 22, 13]. Despite numerous advantages, LiDAR surveys are costly due to the specialist equipment and skilled analysts requires to interpret it [24]. An alternative technology for tree management and monitoring is the use of hyperspectral and remote sensing satellite images. These techniques have advanced significantly over the last couple of decades and are now able to produce high-resolution images which facilitates individual tree crown detection and species classification [8, 18, 6]. A limited number of studies are looking into urban tree classification using RGB aerial images [27, 30]. The study by Wegner & Branson [30] have proposed a CNN based system to catalogue and classify urban trees using publicly available Google satellite images. Their model has been trained and tested on tree crown images from the Pasadena region of California. In another research, Nezami [21] achieved 97% accuracy in tree classification using aerial images and convolutional Neural Networks that testifies the effectiveness of these approaches. Despite staggering accuracy, this study focused on only 3 species which is limited and less practical in the real-world.

The aim of this study is to first generate a labelled tree species dataset of aerial images to facilitate detection, classification and localization of urban trees using publicly available Google Maps aerial images and *Trees In Camden* inventory to supply with GPS locations (Latitude and Longitude) , diameter and species information for trees. This study also aims to assess performance of various state of the art pre-trained deep convolutional neural networks including VGG19, ResNet50, DenseNet121 and InceptionV3 in tree species classification under various training scenarios and parameters. The next section of this paper outlines the dataset preparation process.

2 Dataset Generator Framework

This study proposes a Dataset Generator framework, designed to generate labelled dataset of tree species using aerial RGB images and any given tree inventory to supply species information, GPS coordinates (Latitude and Longitude) and tree diameter. This study uses Google Maps static API to supply aerial RGB images which is a quick and cost effective way of image data collection. This method is especially useful for urban trees as Google offers aerial images with significantly high quality in urban areas. To supply species information, GPS coordinates and tree diameter, this study uses *Trees In Camden* inventory which contains over 23,000 locations of Council owned trees on highways and in parks and open spaces in London Borough of Camden. Each data point contains tree species, height, spread, diameter at breast height (DBH), and maturity [5].

While this inventory consists of hundreds of different tree species, this study only investigates top 6 species with highest frequencies including Ash, Silver Birch, Common Lime, London Plane, Norway Maple and Sycamore. The data is split into subsets with 70% for training, 20% for validation and 10% reserved as unseen test data. The proportional representation of each species is preserved across the subsets so that any class imbalance is retained at each stage. The latitude and longitude co-ordinates of each tree were used as the centre point for each aerial image. A patch of 600x600 with the zoom level of 20 covers large enough area to contain any tree in *Trees In Camden* inventory. A 2D Gaussian kernel which is centred to the tree's GPS coordinates and expands across tree's diameter has been used to generate the ground-truth density maps. Tree images along with density maps will be used to train tree species classification and localisation deep models however this study only focuses on the classification issue. Figure 1 shows some examples of urban tree images with their corresponding ground-through density maps. Since, the number of data samples in the training set is fairly limited and traditionally convolutional neural networks require a very large number of data samples for effective training process. In order to tackle this issue and increase the size and variation of the training data, image augmentation techniques including Rotation, width and height shift, horizontal flip, zoom and brightness are used, thus artificially expanding the size of the training set with new, plausible examples as shown in the Figure 2 [14].

Fig. 1. Examples of urban tree images with their corresponding ground-through density maps.

3 Tree Species Classification

The project explores performance of various state of the art deep convolutional neural networks including VGG19, ResNet50, DenseNet121 and InceptionV3 in tree species classification. Each model was trained with three different training configurations including fully pre-trained, fine-tuning and training from scratch. In all cases, the top fully connected classification layers are modified to accommodate the 6 tree species of our dataset. All the models in this study are trained and tested based on the training, validation and testing sets shown in the Figure 3. This study also investigates the possibility of a reliable tree localisation

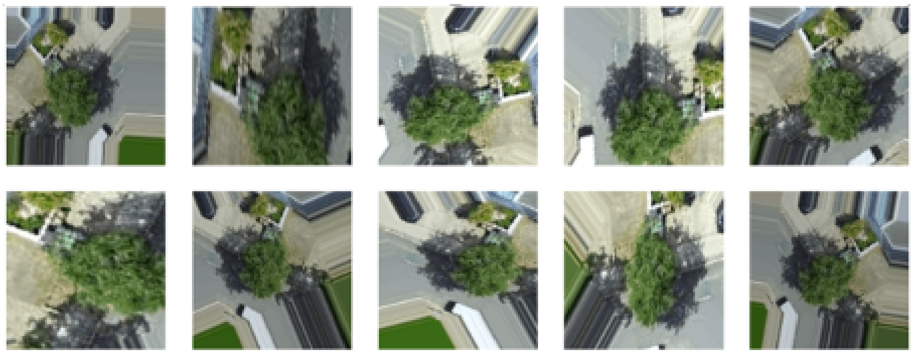
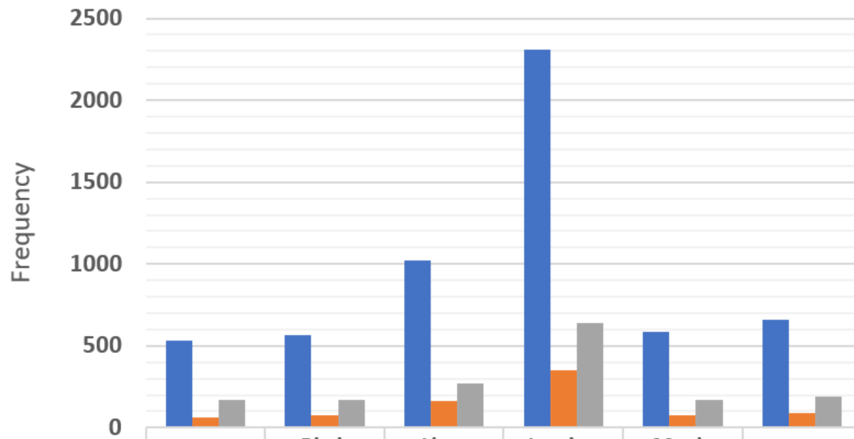


Fig. 2. Examples of the augmentation applied to images in the training data subset

through class activation mapping which demonstrates the discriminative region of the image, which influenced the deep learning model to make the decision [34]. Other training parameters including learning rate, learning decay, loss function, batch size and optimiser are held constant across all models and training configurations. Training samples are shuffled prior to the training process to avoid possible skew toward a certain class and to ensure uniform distribution of classes (tree species) across batches and maximise the information gain per iteration. We used categorical cross-entropy loss function across all models in this study while optimiser of choice is set to Adam. The maximum number of epochs is set to 200 and call-backs are implemented to monitor the validation loss in order to stop the model training if the loss has failed to improve after 10 consecutive epochs. During training iterations, the model with the least amount of loss is saved and used as a benchmark for comparison with other models.

3.1 VGG19 Model

VGG19 [28] is arguably one of the most popular CNN models in image classification. This was the chosen model in similar tree species classification studies by Branson, et al. [2] and Lang [15]. Hence, this study adopted the VGG19 model as it is likely to yield desirable results. Three different training configurations including pre-trained, fine-tuning and training-from-scratch are used to train the VGG19 model. In pre-trained (ImageNet) configuration, weights and biases across all convolutional blocks (feature extractors) are frozen while fully connected layers have been reshaped and retrained to accommodate the 6 tree species of our dataset. In fine-tuning configuration, we have adopted two methods, the first of which unfreezes and retrains 4th and 5th convolutional blocks while keeping the first three blocks frozen. The second method only unfreezes and retrains the last (5th) convolutional block. Similar to pre-trained configuration, fully connected layers have been reshaped and retrained to accommodate the 6 tree species of our dataset. In training-from-scratch configuration, weights



	Ash	Birch Silver	Lime Common	London Plane	Maple Norway	Sycamore
Train	532	565	1024	2311	589	662
Test	64	75	166	348	79	92
Validation	167	167	272	642	173	193

Fig. 3. Training, validation and testing sets counts across top 6 species in Camden dataset

and biases across all convolutional and fully connected layers are initialised using *Glorot_uniform* algorithm. Regardless of training configuration, categorical cross-entropy loss function and Adam optimiser are used to train the VGG19 model.

3.2 ResNet50 Model

ResNet, short for Residual Networks is a classic neural network used as a backbone for many computer vision tasks. The fundamental breakthrough with ResNet was it allowed us to train extremely deep neural networks with 150+ layers without facing problems like vanishing gradients. ResNet uses skip-connections that allows gradients to flow easily from layer to layer and helps even the deepest layer receive activations from the top layers. ResNet50 model is chosen in many similar tree classification studies including [19, 3]. Analogous to VGG19, pre-trained, fine-tuning, and training-from-scratch configurations used to train ResNet50 Model. In fine-tuning configuration, all layers prior to *conv5_block2_add* remained frozen while the subsequent layers have been unfrozen and fine-tuned. Also, the last fully connected dense layer has been reshaped to accommodate the 6 tree species of our dataset. In training-from-scratch configuration, weights and biases across all convolutional and dense layers are initialised using *Glorot_uniform* algorithm. In pre-trained (ImageNet) configuration, we have only reshaped and retrained the last dense layer to accommodate the 6 tree species of our dataset. Categorical cross-entropy loss function and Adam optimiser used to train the ResNet50 Model across all training configurations.

3.3 DenseNet121 Model

DenseNet model is similar to ResNet with some structural difference. ResNet uses addition (+) that merges the previous layer (identity) with the future layer, whereas DenseNet concatenates (.) the output of the previous layer with the future layer. DenseNets connects all layers with matching feature-map sizes directly with each other. To preserve the feed-forward nature, each layer obtains additional inputs from all preceding layers and passes on its own feature-maps to all subsequent layers[10]. DenseNet aims to address vanishing gradients with significantly lesser number of parameters compared to ResNet. DenseNet model was employed in many similar tree classification studies [20, 16]. Similar to VGG19 and ResNet50 models, pre-trained, fine-tuning, and training-from-scratch configurations are used to train DenseNet121 Model. Pre-trained and training-from-scratch configurations are both using similar training strategy and parameters as previous models while fine-tuning configuration, unfreezes and fine-tunes layers subsequent to *conv5_block15_concat* and reshapes the final dense layer to accommodate the 6 tree species of our dataset. Similar to previous models, categorical cross-entropy loss function and Adam optimiser are used to train the DenseNet121 Model across all training configurations.

3.4 InceptionV3 Model

InceptionV3 is the third evolution of Inception architectures family by Google. Inception v3 mainly focuses on lowering computational power by modifying the previous Inception architectures. InceptionV3 model features techniques like factorized convolutions, regularization, dimension reduction, and parallelised computations which set it apart from the competition [29]. Several tree classification studies including [26, 23] employed InceptionV3 model which urged us to explore its efficiency in this research. Similar to previous models, pre-trained, fine-tuning, and training-from-scratch configurations are used to train InceptionV3. In fine-tuning configuration, all layers prior to *mixed9* remained frozen while the subsequent layers have been unfrozen and fine-tuned. In other words, we have attempted to retrain the last Inception module. Moreover, the top fully connected dense layer has been reshaped to accommodate the 6 tree species of our dataset. Pre-trained and training-from-scratch configurations are both using similar training strategy and parameters as previous models. Similar to previous models, categorical cross-entropy loss function and Adam optimiser are used to train the InceptionV3 Model across all training configurations. Also, unlike other models in this study, InceptionV3 has been trained on input image size of 299x299x3. Hence, we made the necessary adjustments in the pre-processing and training process to address this issue.

3.5 Proposed Model

4 Results and Discussion

The training, validation and testing process are performed using the 6 tree species including Ash, Silver Birch, Common Lime, London Plane, Norway Maple and Sycamore. The VGG19, ResNet50, DenseNet121, and InceptionV3 have been trained in three different configurations including pre-trained, fine-tuning (FT), and training-from-scratch (TFS). Categorical cross-entropy has been employed as the loss function of choice across all experiments in this study. All models in this study use Adam optimiser with initial learning rate of 1e-2 and scheduled exponential decay to lower the learning rate as the training progresses. Batch size of 32 has been used across all experiments.

The VGG19 with over 140 million parameters is the most computationally expensive model in this study and consequently took the longest to train and fine-tune. The VGG19 model with fine-tuned 4th and 5th convolutional blocks achieved an accuracy of 71.84 and F1-score of 0.626, outperformed other training configurations of VGG19 model with a reasonable margin. Freezing the 4th convolutional block led to a slight reduction across majority of the evaluation metrics. Freezing the entire convolutional blocks (pre-trained configuration) further reduced the model performance. This indicates textural and geometric features of Google Map's aerial images of urban trees are slightly different to ImageNet's and fine-tuning can positively impact the model performance. Experimental results also show that training-from-scratch (TFS) consistently outperformed other configurations possibly due to lack of training samples. The performance measures obtained by the VGG19 model are recorded in Table 1. Figure 4 shows confusion matrices of different VGG19 training configurations. It appears that regardless of the training configuration, VGG19 struggles at identifying Ash tree species.

Table 1. Evaluation metrics for different VGG19 training configurations

Model	Loss	Accuracy (%)	Avg Class Precision (%)	Avg Class Recall (%)	F1-Score
VGG-19 (4th,5th FT)	1.08	71.84	0.628	0.633	0.626
VGG-19 (5th FT)	1.12	69.42	0.595	0.600	0.596
VGG-19 (Pre-trained)	1.14	68.08	0.583	0.595	0.585
VGG-19 (TFS)	1.21	66.99	0.565	0.583	0.570

The ResNet50 model with just over 25 million parameters is significantly faster in training and inference. In general, ResNet50 consistently under-performed

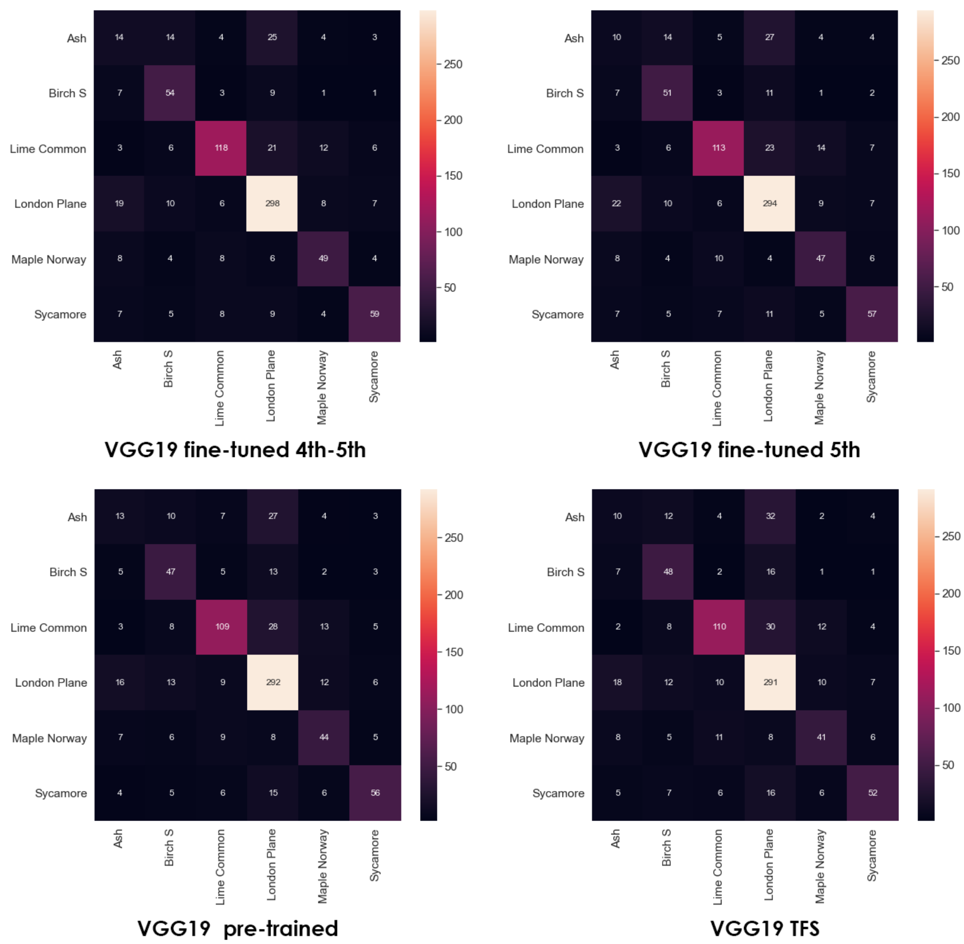


Fig. 4. Confusion matrices for different VGG19 training configurations

VGG19 regardless of training configurations. Just like VGG19, ResNet50's performance topped at fine-tuned training configuration, where the maximum accuracy of 68.93 and F1-score of 0.583 have been registered. We believe ResNet50's performance could be further improved by investigating different fine-tuning (freezing/unfreezing) possibilities. Experiments shows significant drop in ResNet50 performance in both pre-trained and training-from-scratch (TFS) configuration. This is consistent with what has been observed in the VGG19 experiments. We believe insufficient training samples is the main reason of such behaviour. The performance measures obtained by the ResNet50 model are recorded in Table 2. Figure 5 shows confusion matrices of different ResNet50 training configurations.

It appears ResNet50 not only struggles with identification of Ash tree species but also performs poorly at Silver Birch classification.

Table 2. Evaluation metrics for different ResNet50 training configurations

Model	Loss	Accuracy (%)	Avg Class Precision (%)	Avg Class Recall (%)	F1-Score
ResNet50 (FT)	1.12	68.93	0.576	0.601	0.583
ResNet50 (Pre-trained)	1.44	62.01	0.486	0.502	0.491
ResNet50 (TFS)	1.49	61.77	0.486	0.503	0.493

The DenseNet121 with just over 8 million parameters is the lightest and fastest model to train in this research. However, this comes with the cost of performance. At its peak, the DenseNet121 achieved accuracy of 66.55 and F1-score of 0.56 which is considerably lower than its counterparts in this study. Just like ResNet50 and VGG19 the highest performance observed under fine-tuning configuration. DenseNet121 under training-from-scratch (TFS) configuration, registered the lowest accuracy (59.95) and F1-score (0.461) across all the experiments in this study. The performance measures obtained by the DenseNet121 model are shown in Table 3. Figure 6 shows confusion matrices of different DenseNet121 training configurations. Similar to ResNet50, DenseNet121 struggles with classification of Ash and Birch Silver tree species.

Table 3. Evaluation metrics for different DenseNet121 training configurations

Model	Loss	Accuracy (%)	Avg Class Precision (%)	Avg Class Recall (%)	F1-Score
DenseNet121 (FT)	1.24	66.55	0.548	0.570	0.560
DenseNet121 (Pre-trained)	1.59	60.32	0.470	0.473	0.468
DenseNet121 (TFS)	1.67	59.95	0.465	0.463	0.461

Last but not least, InceptionV3 model with over 23 million parameter is the second fastest model in this study. However unlike DenseNet121 (fastest model in this study) its speeds comes with no performance penalty. The InceptionV3 model achieved impressive accuracy of 73.54 and F1-score of 0.646, the highest recorded across all the experiments in this study. We believe InceptionV3's performance could be even further improved by investigating different fine-tuning (freezing/unfreezing) possibilities. The performance measures obtained by the

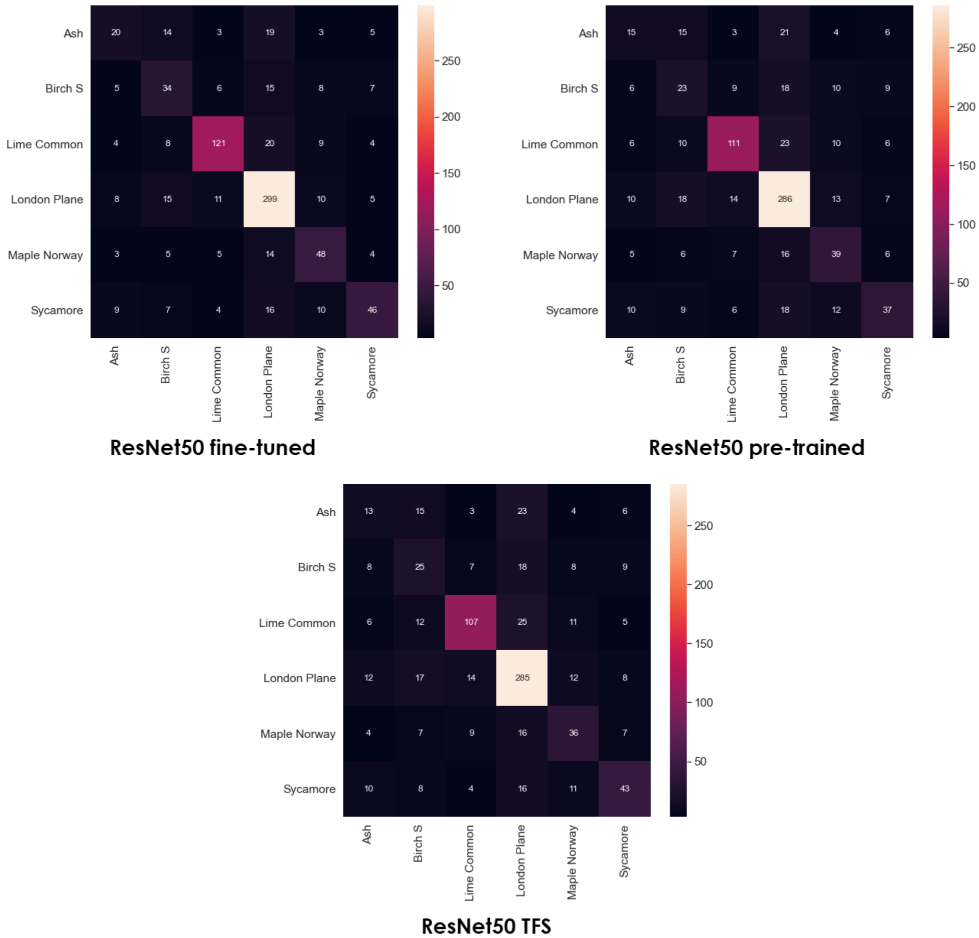


Fig. 5. Confusion matrices for different ResNet50 training configurations

InceptionV3 model are shown in Table 4. Figure 7 shows confusion matrices of different InceptionV3 training configurations. A sensible improvement in classification can be observed across all tree species but similar other models in this study, InceptionV3 struggles with segregation of Ash and London Plane species.

A deeper investigation into model's training behaviours shows that VGG19 suffers from a considerable amount of overfitting. Due to the fact that the VGG19 features huge number of parameters (140 million), training with small datasets like *Trees In Camden* leads to issues like overfitting. This can be slightly mitigated by introduction of Regularisation and Dropouts into the model. In general, we have realised reported performance measures across all experiments were effected by insufficient training samples. One possible solution is to combine im-

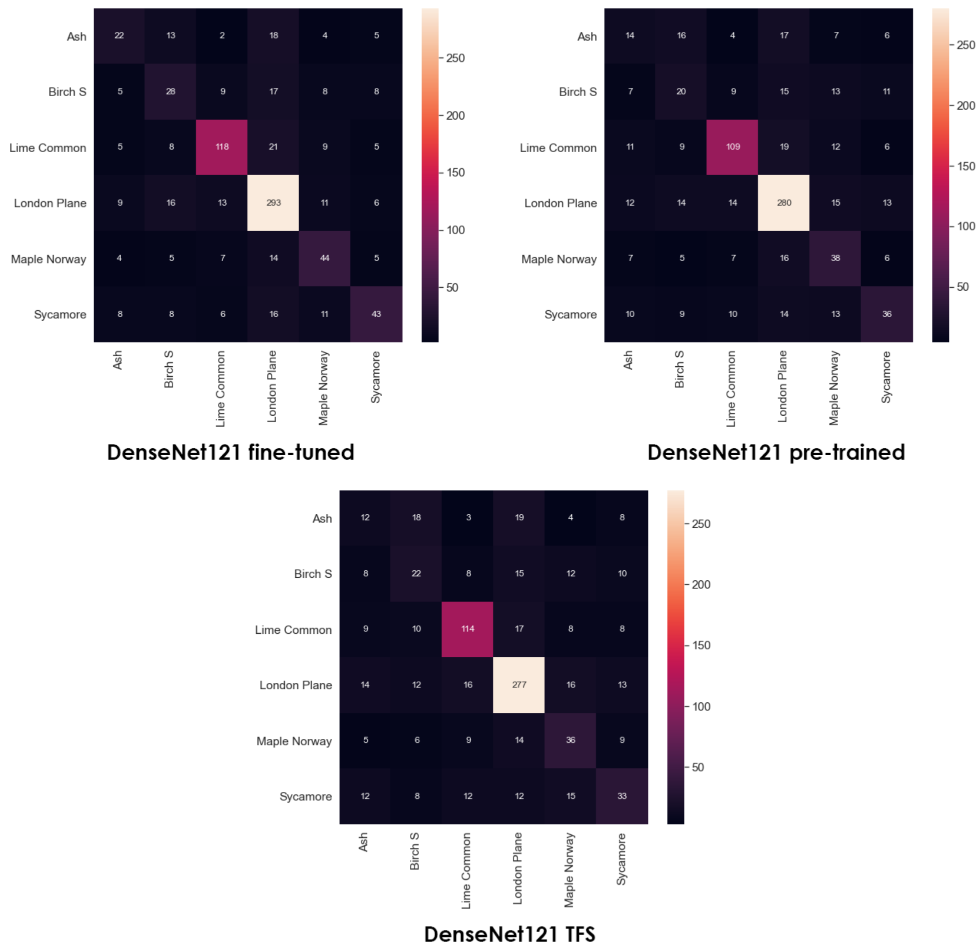


Fig. 6. Confusion matrices for different DenseNet121 training configurations

ages from other repositories such as *Pasadena Urban Trees* [31]. An in-depth investigation into other possible fine-tuning configurations could also mitigate this issue. It is worth mentioning that some images in our dataset may contain more than one tree if they are situated close together and, depending upon the accuracy of the location data, the labelled tree may not necessarily be accurate. Although the Camden tree inventory contains a large amount of detailed data, improvements could be made to ensure that common name species labels are correct. For example, *Maple – Crimson King Norway* is a separate category to *Maple – Norway*. Combining these would increase the number of images in *Maple–Norway* class by 9%. Similarly, *Wier-maple* and *Maple–Silver* are distinct categories, however the *weir-maple* is a type of *silver-maple* and so grouping

Table 4. Evaluation metrics for different InceptionV3 training configurations

Model	Loss	Accuracy (%)	Avg Class Precision (%)	Avg Class Recall (%)	F1-Score
InceptionV3 (FT)	0.96	73.54	0.650	0.650	0.646
InceptionV3 (Pre-trained)	1.09	69.90	0.596	0.595	0.595
InceptionV3 (TFS)	1.20	66.14	0.540	0.550	0.548

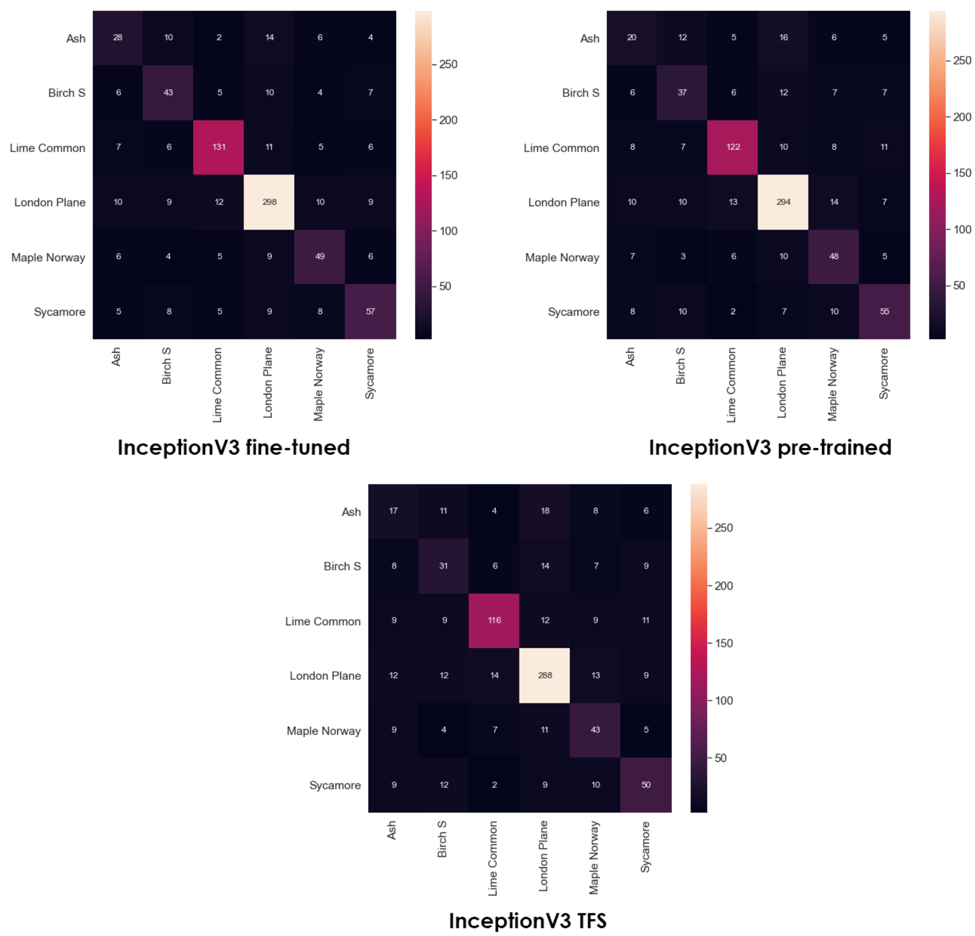


Fig. 7. Confusion matrices for different InceptionV3 training configurations

these together would triple the size of this class. Also, we have realised imbalance nature of our dataset adversely impacted the results. The attempt to add class weights to account for the data imbalance could be a possible mitigation plan. Further research into handling imbalanced data could be conducted to reduce bias towards the larger class (London Plane). One such method for this would be to over or under sample the training images to create balance in this data set [11].

5 Conclusion

Tree detection and species classification using aerial or satellite imagery was an inherently expensive and time-consuming task. This research examined the possibility of urban-tree detection and species classification using Google Maps aerial images and publicly available tree inventories such as *Trees In Camden* to supply GPS coordinates and tree species information. This can significantly reduce the cost of surveying and data collection and overall helps to leverage effective forest and urban tree management. The work involved investigating several state of the art deep convolutional neural network models including VGG19, ResNet50, DenseNet121 and InceptionV3 at three different training configurations including fully pre-trained, fine-tuning and training from scratch. Results shows, a fine-tuned InceptionV3 model is able to classify up to 6 different species with over 73% accuracy and 0.646 F1-score. While this is far from an ideal solution, this study shows the possibility of urban-tree species classification using free and publicly available Google Map's images. Future work such as investigating other known popular models such as AlexNet, InceptionResNetV2 and Xception or other possible fine-tuning configurations could likely to improve the metrics.

References

[1] L. Baeten and H. Bruelheide. "Identifying the tree species compositions that maximize ecosystem functioning in European forests." In: *Journal of Applied Ecology* 56.3 (2018).

[2] S. Branson et al. "From Google Maps to a Fine-Grained Catalog of Street trees". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 135 (2018).

[3] Kaili Cao and Xiaoli Zhang. "An improved res-unet model for tree species classification using airborne high-resolution images". In: *Remote Sensing* 12.7 (2020), p. 1128.

[4] K. Chandler et al. "Influence of tree species and forest land use on soil hydraulic conductivity and implications for surface runoff generation". In: *Geoderma* 310 (2017), pp. 120–127.

[5] London Borough of Camden Coucil. *Trees In Camden: Open Data Portal.* 2021. URL: https://opendata.camden.gov.uk/Environment/Trees-In-Camden/csqp-kdss.

[6] Michele Dalponte, Lorenzo Frizzera, and Damiano Gianelle. "Individual tree crown delineation and tree species classification with hyperspectral and LiDAR data". In: *PeerJ* 6 (2019), e6227.

[7] G. H. Donovan, S. Landry, and C. Winter. "Urban trees, house price, and redevelopment pressure in Tampa". In: *Florida. Urban Forestry & Urban Greening* 38 (2019), pp. 330–336.

[8] G. A. Fricker et al. "A Convolutional Neural Network Classifier Identifies Tree Species in Mixed-Conifer Forest from Hyperspectral Imagery". In: *Remote Sensing* 11.19 (2019).

[9] L. Gamfeldt and T. Snall. "Higher levels of multiple ecosystem services are found in forests with more tree species." In: *Nature Communications* 4 (2013).

[10] Gao Huang et al. "Densely connected convolutional networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2017, pp. 4700–4708.

[11] Van Hulse et al. *Experimental perspectives on learning from imbalanced data*. ICML '07: Proceedings of the 24th international conference on Machine learning, 935-942; https://doi.org/10.1145/1273496.1273614, 2007.

[12] S. Kim et al. "Individual tree species identification using LiDAR intensity data." In: *Portland: ASPRS 2008 Annual Conference.* (2008).

[13] Barbara Koch, Ursula Heyder, and Holger Weinacker. "Detection of individual tree crowns in airborne lidar data". In: *Photogrammetric Engineering & Remote Sensing* 72.4 (2006), pp. 357–363.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton. "ImageNet classification with deep convolutional neural networks". In: *Communications of the ACM* 60.6 (2017), pp. 84–90.

[15] N. Lang. "Deep learning and Google Maps for tree monitoring." In: (2020).

[16] Hui Li et al. "CNN-Based Tree Species Classification Using Airborne Lidar Data and High-Resolution Satellite Image". In: *IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium.* IEEE. 2020, pp. 2679–2682.

[17] L. Manickathan et al. "Parametric study of the influence of environmental factors and tree properties on the transpirative cooling effect of trees". In: *Agricultural and Forest Meteorology* 248 (2017), pp. 259–274.

[18] Julia Maschler, Clement Atzberger, and Markus Immitzer. "Individual tree crown segmentation and classification of 13 tree species using airborne hyperspectral data". In: *Remote Sensing* 10.8 (2018), p. 1218.

[19] S Natesan, C Armenakis, and U Vepakomma. "RESNET-BASED TREE SPECIES CLASSIFICATION USING UAV IMAGES." In: *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences* (2019).

[20] Sowmya Natesan, Costas Armenakis, and Udayalakshmi Vepakomma. "Individual tree species identification using Dense Convolutional Network (DenseNet) on multitemporal RGB images from UAV". In: *Journal of Unmanned Vehicle Systems* 8.4 (2020), pp. 310–333.

[21] S. Nezami et al. “Tree Species Classification of Drone Hyperspectral and RGB Imagery with Deep Learning Convolutional Neural Networks.” In: *Remote Sensing* 12.7 (2020).

[22] Mats Nilsson. “Estimation of tree heights and stand volume using an airborne lidar system”. In: *Remote sensing of environment* 56.1 (1996), pp. 1–7.

[23] Masanori Onishi and Takeshi Ise. “Automatic classification of trees using a UAV onboard camera and deep learning”. In: *arXiv preprint arXiv:1804.10390* (2018).

[24] Rezatec. “Satellites vs. Lidar for Forestry Management?” In: (2020).

[25] S. Rust. “Tree inventory, risk assessment and management”. In: *Urban Tree Management : For the Sustainable Development of Green Cities*. Ed. by A. Roloff. Gottingen: John Wiley & Sons, 2016, pp. 178–210.

[26] Anastasiia Safonova et al. “Detection of fir trees (Abies sibirica) damaged by the bark beetle in unmanned aerial vehicle images with deep learning”. In: *Remote sensing* 11.6 (2019), p. 643.

[27] Lakshmi Babu Saheer and Mohamed Shahawy. “Self-supervised approach for Urban Tree Recognition on Aerial Images”. In: *Proceedings of the 17th International conference on Artificial Intelligence Applications and Innovations*. 2021.

[28] K. Simonyan and A. Zisserman. “Very deep convolutional networks for large-scale image recognition”. In: *arXiv:1409* 1556. (2014).

[29] Christian Szegedy et al. “Rethinking the inception architecture for computer vision”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2818–2826.

[30] J. D. Wegner et al. “June)”. In: *Cataloging Public Objects Using Aerial and Street-Level Images – Urban Trees. Retrieved May* 1 (2020). URL: http://www.vision.caltech.edu/publications/CVPR2016-WegnerBransonEtAl.pdf.

[31] Jan D Wegner et al. “Cataloging public objects using aerial and street-level images-urban trees”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 6014–6023.

[32] P. Wilkes et al. “Estimating urban above ground biomass with multi-scale LiDAR”. In: *Carbon Balance Manage* 13.10 (2018).

[33] K. L. Wolf. “Business District Streetscapes, Trees, and Consumer Response”. In: *Journal of Forestry* 103.8 (2005), pp. 396–400.

[34] Bolei Zhou et al. “Learning deep features for discriminative localization”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2921–2929.