

# Hidden features: Experiments with feature transfer for fine-grained multi-class and one-class image categorization

Varvara Vetrova      Sheldon Coup

*School of Mathematics and Statistics*

*University of Canterbury*

Christchurch, New Zealand

{varvara.vetrova@sheldon.coup@pg.}canterbury.ac.nz

Eibe Frank

*Department of Computer Science*

*University of Waikato*

Hamilton, New Zealand

eibe@cs.waikato.ac.nz

Michael J. Cree

*School of Engineering*

*University of Waikato*

Hamilton, New Zealand

cree@waikato.ac.nz

**Abstract**—Can we apply out-of-the box feature transfer using pre-trained convolutional neural networks in fine-grained multi-class image categorization tasks? What is the effect of (a) domain-specific fine-tuning and (b) a special-purpose network architecture designed and trained specifically for the target domain? How do these approaches perform in one-class classification? We investigate these questions by tackling two biological object recognition tasks: classification of “cryptic” plants of genus *Coprosma* and identification of New Zealand moth species. We compare results based on out-of-the-box features extracted using a pre-trained state-of-the-art network to those obtained by fine-tuning to the target domain, and also evaluate features learned using a simple Siamese network trained only on data from the target domain. For each extracted feature set, we test a number of classifiers, e.g., support vector machines. In addition to multi-class classification, we also consider one-class classification, a scenario that is particularly relevant to biosecurity applications.

In the multi-class setting, we find that out-of-the-box low-level features extracted from the generic pre-trained network yield high accuracy (90.76%) when coupled with a simple LDA classifier. Fine-tuning improves accuracy only slightly (to 91.6%). Interestingly, features extracted from the much simpler Siamese network trained on data from the target domain lead to comparable results (90.8%). In the one-class classification setting, we note high variability in the area under the ROC curve across feature sets, opening up the possibility of considering an ensemble approach.

**Index Terms**—fine-grained categorization, feature transfer, one-class classification, Siamese networks

## I. INTRODUCTION

Feature transfer is a simple approach to training classification models for new categorization tasks by transferring knowledge from a source domain to the target domain. It has proven to be surprisingly effective, particularly in the context of image classification. Features extracted from a neural network that has been trained on a sufficiently large and diverse source collection of labeled images can enable a simple classifier such as a linear support vector machine to achieve highly competitive classification accuracy in the target

domain [1]. This is particularly useful in situations where little labeled data is available in the target domain [2], [3].

In this paper, we tackle the challenge of multi-class and one-class classification of fine-grained datasets, where accurate recognition of fine detail is important to discriminate between classes of images and thus achieve good classification performance [4]. Fine-grained recognition tasks differ from general image classification tasks because variation between images pertaining to different classes is much smaller. Examples of fine-grained classification problems include identification of cars, aircrafts, and faces [5], [6].

There is some evidence [7], obtained on a fine-grained multi-class benchmark dataset involving the identification of bird species, that feature transfer can be applied successfully in this difficult scenario. Here, we add to this body of evidence by providing experimental results for multi-class and one-class classification obtained on two fine-grained species identification problems: automatic classification of plants of the genus *Coprosma*, and discrimination of species of New Zealand native and invasive moths. These are challenging fine-grained tasks because it is often difficult to distinguish species of biological organisms solely based on morphology. In particular, plants of genus *Coprosma* that occur in New Zealand contain species that are hard to differentiate even by expert botanists; some species of moth present a similar level of difficulty.

We study the classic approach, first advocated in [1], where the network used for feature extraction stems from a very large generic image classification problem and is not adapted at all by adjusting its parameters to the target domain. We also consider two alternatives: (a) fine-tuning this network using a collection of images from the target domain before feature extraction on other images from this domain, and (b) training a—much simpler—feature extraction network from scratch for the target domain. In the latter case, we use a Siamese network trained using triplets of images. For reference, we also include results obtained by training a state-of-the-art network for the target data and applying this network directly to classify test images in the target data.



Fig. 1. Plant of genus *Coprosma*, *Crassifolia* species.

Our results show that fine-tuning the preexisting generic state-of-the-art network to the target domain yields little benefit. However, the features extracted using the Siamese network yield performance that is competitive to that obtained using features from the much more complex generic network. In the one-class classification task, performance varies depending on the particular species that is held out for testing: in some cases, the domain-specific Siamese network yields better results; in others, the generic network proves more beneficial. This raises the interesting question of whether results can be further improved in the future by combining the sets of features obtained using the two models.

The paper is organised as follows. Section II discusses the two fine-grained species classification problems we consider and presents details of the datasets we use. Section III describes the network architectures and learning approaches we evaluate. Section IV presents experimental results comparing classification performance on the two classification problems we consider. Section V has some concluding remarks.

## II. DATASETS

The two classification domains considered are both specific to New Zealand: classification of species of *Coprosma* that occur in New Zealand and discrimination between New Zealand native and invasive moths. The first classification problem is primarily of interest to botanists: there is only a very small number of experts who are able to accurately discriminate between *Coprosma* species, so an automatic method of classification would be of great value. The second classification problem has potential applications in biosecurity because New Zealand’s horticulture and forestry industry could be heavily affected by the introduction of certain species of moths.

### A. Plants of genus *Coprosma*

There are 17 species of *Coprosma* represented in our *Coprosma* dataset. For each species, there are one to eight plants (mean 4.9) for a total of 83 plants, and there are typically ten images per plant (each of a different branch of the plant, mean 9.9), for a total of 819 images such as the one shown in Figure 1. Each image contains  $5184 \times 3456$  pixels and is stored as a high quality JPEG. Images were taken by placing



Fig. 2. Moth specimen, *Nyctemera annulata* x *amicus*.

each branch on a black background. A ruler was present in the top left-hand corner of each image to indicate scale, which we removed before further processing occurred by setting the pixels of the ruler to black. 3799 non-overlapping crops of size  $1024 \times 1024$  pixel were extracted from the images to form our dataset for machine learning.

### B. New Zealand native and invasive moths

Our primary moth dataset contains 10 species of moth. It includes both, moths native to NZ and potential invaders. For each species, the data contains between 2 and 54 images of individual specimens (mean of 20). Hence, the amount of data available per category is very imbalanced. All of the images have a set width of 1181 pixels and varying height, and were taken from a straight-on dorsal view of the specimens. In each image, a scale bar was present originally in the bottom-left corner but has been removed for training and testing.

The dataset contains two subgroups of species that are particularly difficult to distinguish based solely on the visual information available for classification. The first subgroup contains three species of the genus *Nyctemera*: *Nyctemera amicus*, *Nyctemera annulata*, and a hybrid of the two. An example hybrid is shown in Figure 2. *Nyctemera annulata* is endemic to NZ, *Nyctemera amicus* has arrived in northern NZ from Australia, and these two species interbreed. The second subgroup of visually highly similar species in the data comprises the two species *Utetheisa pulchelloides* and *Utetheisa lotrix*. Both species are known to migrate to NZ.

The other five species in this moth dataset are *Teia nartoides*, *Lymatria dispar* and *Orgyia thyellina* —all three exotic pest species that are not established in NZ—*Tyria jacobaeae*, which was introduced to NZ as a biocontrol agent, and *Cebysa leucotelus*, an Australian species that is now established in NZ.

### C. Larger moths of New Zealand

In addition to our primary moth dataset (B), we have prepared a second dataset of NZ moths based on images publicly available from Landcare Research [8]. We use this second dataset, which contains 1326 images distributed across eight families of moth, to train domain-specific feature extraction

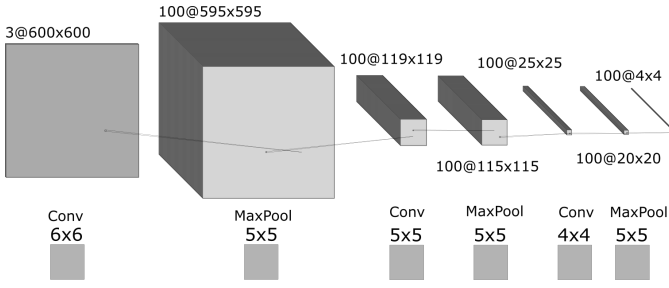


Fig. 3. Siamese network architecture used to learn embedding based on triplets of moths.

networks that are used to construct feature vectors for our primary Moth dataset (B). The images have been padded to 600x600 pixels and scale information has been removed. Of the 10 species in the primary dataset, the *Nyctemera* species, the *Uteteisa* species, *Teia nartoides*, *Orgyia thyellina*, and *Tyria jacobaeae*, are also present in this larger dataset.

For many species of moth in this second dataset, there are only two images: one for a male specimen and one for a corresponding female specimen. In total, there are 495 of these pairs. We use this subset of 990 images to form triplets of images that are used to train a Siamese network. As a second domain-specific feature extractor, from the full dataset, we train a multi-class network by learning to discriminate the 1326 images into families.

### III. METHODS

We performed two broad categories of experiments: multi-class classification and one-class classification. Features extracted from generic and domain-specific convolutional neural networks formed the basis of these experiments.

#### A. Feature extraction

We first discuss the basic feature extraction methods.

1) *Siamese network embeddings*: We first discuss feature training using Siamese neural networks [9], an approach inspired by the obvious similarity between species identification and face recognition: generally, few images are available for each class, and the differences between images of the same class are small. Siamese networks have been used extensively to address challenges in face recognition and one-shot classification [10], [11]. They consist of two blocks of identical convolutional neural networks joined by a loss function that encourages the two networks to generate similar output when each receives similar input (i.e., two images of the same species) and distinct output when each receives different input (i.e., two images of different species). The parameters of models based on Siamese networks are similar to other types of deep learning models and include choice of architecture and loss function. In order to learn features, we use an architecture consisting of one parallel (twin) block with three convolutional layers in combination with the triplet margin ranking loss [12], applied with the value one for the margin parameter:

- Convolutional layer:- stride 1, 100 filters,  $f = 6$

- MaxPool layer:- stride 1,  $f = 5$
- ReLu activation function
- Convolutional layer:- stride 1, 100 filters,  $f = 5$
- MaxPool layer:- stride 1,  $f = 5$
- ReLu activation function
- Convolutional layer:- stride 1, 100 filters,  $f = 4$
- MaxPool layer:- stride 1,  $f = 5$

This architecture is illustrated in Figure 3. Thus, the input of the network is an image of size 600x600 and the output is an embedding vector of size 1600x1. We trained this Siamese network on the part of the dataset C that consists of one pair of male and female specimen per species. Triplets of images were prepared in mini-batches of size 15 for gradient descent. An example batch with only five triplets is shown in Figure 4. Triplets were obtained in the following manner. First, a male/female pair was picked at random from the 495 available pairs. Then, either the male or the female moth was selected at random to serve as the anchor image for the triplet. The remaining image from the pair became the positive (i.e., matching) example. To complete the triplet with an image that does not match, an image from a different male/female pair in the dataset was picked at random. Once trained, the 1600-dimensional embeddings extracted with this Siamese network served as inputs for classification algorithms described in Sections III-C, III-D.

2) *Feature extraction from InceptionV3 architecture*: We also utilized features extracted from the state-of-the-art InceptionV3 network pre-trained on the ImageNet dataset. We considered using the network directly, without any further adjustments, but also generated results obtained using fine-tuning to the target domain. The fine-tuning approach is discussed in Section III-B. In both cases, features were extracted in the following way. First, a forward pass of the InceptionV3 network was performed on the images from the respective datasets. Then, the 2048-dimensional vector output from the final pooling operator in the network was extracted. We also evaluated augmenting this feature set with lower-level features extracted from the network. To this end, the final vector representation of each image was found by concatenating the above 2048-dimensional feature vector with an additional feature vector extracted from an earlier layer. As the InceptionV3 network is comprised of several “Inception Blocks” in sequence, the additional feature vector was extracted between these blocks. The extractions were of varying 3D shapes, therefore the maximum response was taken from each filter to convert it to vector form. Principal component analysis was applied to the extracted vectors in order to reduce dimensions to 128. The resulting features served as inputs for the classification algorithms described in in Sections III-C, III-D.

#### B. Fine-tuning of convolutional neural networks

In order to test whether fine-tuning on a similar domain affects classification accuracy when using the extracted features, the InceptionV3 network pre-trained on ImageNet was fine-tuned on dataset C to generate features for dataset B. This also provides a direct comparison to the Siamese network

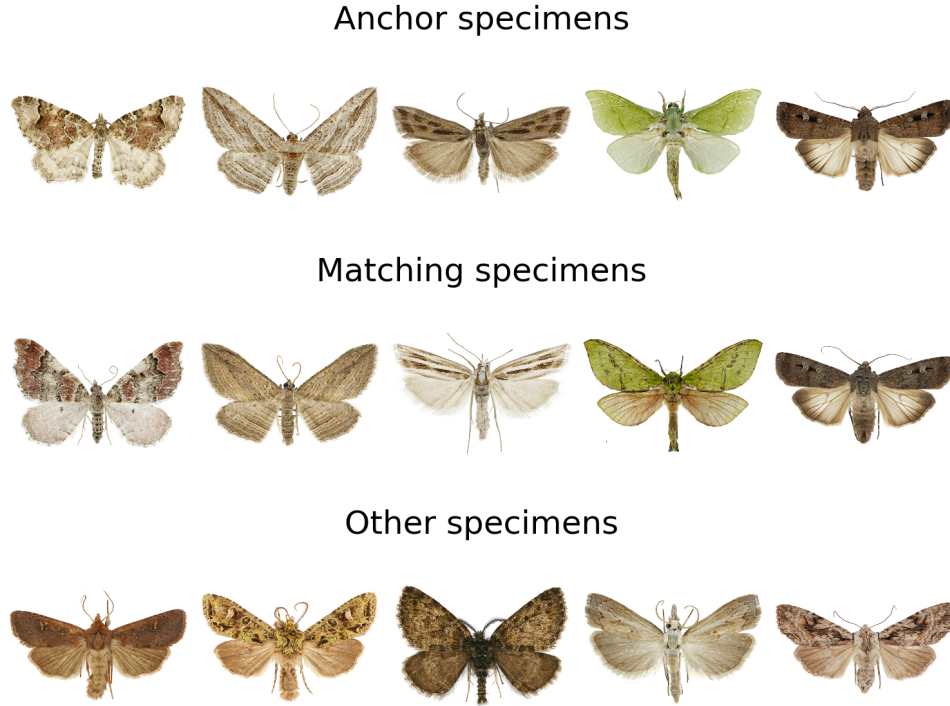


Fig. 4. Mini-batch of triplet-images of size 5. The upper row contains anchor images, the middle row shows matching images and the lower row exhibits negative examples. Images obtained from [8].

approach, which was trained from scratch on dataset C and used to extract features from dataset B. Two variants of fine-tuning were considered:

- First 5 modules frozen and not modified during fine-tuning; and
- all layers unfrozen.

We utilized weight freezing here in order to evaluate the effect on accuracy of combining generic and fine-tuned features. Features were extracted from different layers as described above in Section III-A2. To learn the features, dataset C was used to form a multi-class classification problem by classifying moths at the genus level.

A natural question that arises is how well a state-of-the-art network fine-tuned to the target domain performs when used directly for classification, without applying feature extraction and application of a second machine learning approach such as a support vector machine to the extracted features. To investigate this, we used a convolutional neural network based on the InceptionResNetV2 architecture, also pre-trained on the Imagenet dataset, that we fine-tuned to the two respective datasets A and B. The results were taken to form a baseline in our multi-class classification experiments. Taking into account the relatively small number of images in the datasets, 5-fold cross-validation was used to estimate model performance. The following settings were utilized in these experiments:

- Coprosma dataset A: 12000 steps per run, learning rate of 0.01, learning rate decay factor of 0.94, learning rate decaying every 5 epochs.

- Moth dataset B: 5000 steps per run, learning rate of 0.1, learning rate decay factor of 0.97, learning rate decaying every 15 epochs.

#### C. Multi-class classification

Multi-class classification was performed on both the Coprosma and the moth data, classifying at the species level. We evaluated the following approaches:

- direct classification based on fine-tuning of the Inception-ResNetV2 network—the base-line approach; and
- classification using features extracted from the InceptionV3 network and Siamese embeddings as inputs for the following classifiers: SVMs with linear and RFB kernels, KNN, MLP, ExtraTrees, LDA and GNB, as implemented in scikit-learn [13], using default parameter settings.

In the feature transfer experiments, to obtain a robust estimate of accuracy for each classifier, Monte-Carlo cross-validation was used with 100 random splits of the data into training/testing sets, using 90% of the data for training and the rest for testing. The details of the multi-class experiments we performed are summarized in Table I.

#### D. One-class classification

We performed one-class classification experiments for both the moth and the Coprosma data in the following way. For each run of the experiments, one species was designated as the “novel” class, and the dataset was split into 90% training and 10% test sets. All of the feature vectors associated with

TABLE I  
MULTI-CLASS CLASSIFICATION EXPERIMENTS

Coprosma data (A)	Moth data (B)
–	Direct classification with fine-tuned Inception ResNetV2
–	Classification with features from unmodified InceptionV3
–	Classification with features from Siamese network, learned on dataset C
–	Classification with features from InceptionV3, fine-tuned on dataset C, no frozen layers
–	Classification with features from InceptionV3, fine-tuned on dataset C, first 5 modules frozen

TABLE II  
RESULTS OF MULTI-CLASS CLASSIFICATION, COPROSMA DATASET (A)

Experiment	Percent correct
Direct classification with fine-tuned InceptionResNetV2	74.37
Classification with features from unmodified InceptionV3 (SVM-RBF, mixed5)	78.5

the designated “novel” species were removed from the training set and appended to the test set. Then, a one-class model was trained on the training set. The objective was to learn a model describing all the other species, joined into one class, so that this model could be used to reject instances of the “novel” species in test set. The same sets of features utilized in the multi-class experiments (Table I) were also used as inputs here. To perform one-class classification on the extracted features, we used one-class SVMs with linear and RBF kernels. Using each species as the “novel” species in turn, the area under the ROC curve (AUROC) was calculated, based on the average over 100 random stratified train/test splits.

#### IV. RESULTS

We first discuss the experimental results obtained for multi-class classification and then consider the one-class case.

##### A. Multi-class classification

The results of the multi-class experiments on the Coprosma dataset (A) and the moth dataset (B) are summarized in Tables II and III respectively. The index  $i$  in  $mixed_i$  indicates the block of the InceptionV3 network after which additional features were extracted. We only show results for the best value of  $i$  and the best learning algorithm applied. As can

TABLE III  
RESULTS OF MULTI-CLASS CLASSIFICATION, MOTH DATASET (B)

Experiment	Percent correct
Direct classification with fine-tuned InceptionResNetV2	86.63
Classification with features from umodified InceptionV3 (LDA classifier, mixed0)	90.76
Classification with features from Siamese network (LDA classifier)	90.8
Classification with feature extracted from InceptionV3, fine-tuned on dataset C, no frozen layers (LDA classifier, mixed4)	91.4
Classification with feature extracted from InceptionV3, fine-tuned on dataset C, first 5 modules frozen (LDA, mixed0)	91.6

been seen in Table II, on the Coprosma dataset (A), using features extracted with the generic pre-trained InceptionV3 network gives better accuracy than direct classification using an InceptionResNetV2 network that was fine-tuned to the Coprosma data. Overall, accuracy on this classification task is quite low, which can potentially be explained by the spatial structure of the arrangement into leaves and the different sizes of leaves across different species.

In the case of the moth dataset (B), as shown in Table III, performance across different experiments is very similar, but direct classification using InceptionResNetV2 is again the worst option. It is interesting to note that the features extracted from the Siamese network with a relatively simple architecture (Fig. 3) trained on dataset C perform essentially just as well as the other feature sets that are based on a much more complex state-of-the-art network architecture. The small difference in accuracy is well within the variance of the estimates for this comparably small dataset containing only roughly 200 images. A particular noteworthy outcome of the experiments is that the simple LDA classifier consistently shows the best performance among the seven classifiers. Apart from GNB, the Gaussian naive Bayes classifier, which is presumably too restrictive for this data, the LDA classifier is the only generative modelling approach in our experiments; all the other learning algorithms are discriminative. Our results are consistent with earlier work showing that generative classifiers can be preferable when the size of the training set is limited [14].

The results also show that low-level features obtained from the InceptionV3 network ( $mixed_0$  and  $mixed_4$ ) work well on the moth data. The ability of generic features to be powerful predictors in the case of moth datasets can be attributed to moth morphology: differences between species can be seen in small details of wings and antennae.

It is also worth noting that across all feature transfer experiments, there was a significant disparity between results based on SVMs with a linear kernel and those with an RBF kernel. We observed a difference of around 20%. Considering the good performance of LDA, also a linear classifier, this can be attributed to the implementation of these classifiers in scikit-learn. The non-linear SVM performs pairwise classification while the linear SVM performs one-vs-all classification. The poor performance of pairwise classification can potentially be attributed to the lack of training images in each class and therefore lack of information required to generalise.

##### B. One-class classification

The results of one-class classification on the moth data are summarized in Table IV, showing average AUROC per “novel” species. The columns correspond to results obtained with the following feature sets, obtained by training a one-class SVM with an RBF kernel on these features:

- Column 1: Features extracted from the Siamese network trained on dataset C.
- Column 2: Features extracted from the second module and the last module of InceptionV3.

TABLE IV  
RESULTS OF ONE-CLASS CLASSIFICATION, MOTH DATASET, AUROC

Species/Experiments	1	2	3	4
Cebysa leucotelus	0.84	0.94	0.99	0.93
Lymantria dispar	0.91	0.97	0.91	0.96
Nyctemera amicus	0.69	0.56	0.6	0.49
Nyctemera amicus x annulata	0.84	0.62	0.61	0.62
Nyctemera annulata	0.76	0.69	0.78	0.65
Orgyia thyellina	0.72	0.86	0.92	0.87
Teia anartoides	0.64	0.8	0.76	0.89
Utetheisa pulchelloides	0.58	0.84	0.9	0.87
Utetheisa lotrix	0.26	0.59	0.63	0.7
Tyria jacobaeae	0.92	0.96	0.99	0.93

- Column 3: Features extracted from the second and the last module of InceptionV3 fine-tuned on dataset C, with no layers frozen during fine-tuning.
- Column 4: Features extracted from the first and the last module of InceptionV3 fine-tuned on dataset C, with the first five modules frozen during fine-tuning.

It can be seen that results vary across species, particularly for those groups of species that are difficult to tell apart by humans: the Utethesis species and the Nyctemera species. The feature set obtained using the Siamese network performs much worse than the other feature sets on the former group, but it performs much better on the Nyctemera species. Hence, the different feature sets capture different aspects of the moths. Consequently, it would be of interest to implement experiments based on an ensemble of feature sets by combining classifications obtained using different feature sets.

## V. CONCLUSIONS AND FUTURE WORK

We experimentally tested multi- and one-class classification methods based on features extracted from both generic and fine-tuned convolutional neural networks in a challenging fine-grained setting: specially assembled datasets of Coprosma plants and NZ native and invasive moth. In addition to using the InceptionV3 architecture for feature extraction, we also proposed a simple convolutional network architecture trained as a Siamese network using the triplet margin ranking loss. Regarding our multi-class classification experiments, we observed that feature extraction using either the generic pre-trained network (both target domains) or a network adjusted to the target domain (moth data only), followed by classification using a support vector machine or linear discriminant analysis, improved on performing direct classification using the InceptionResNetV2 network fine-tuned to the target domain. However, in our experiments on the moth data, we did not observe a significant increase in accuracy using features extracted by fine-tuning networks on images from the moth domain instead of features from the generic network. On the other hand, features extracted with the Siamese network approach, with a much simpler architecture trained from scratch in the target domain, achieved comparable results to those from the much more complex InceptionV3 network.

In the case of one-class classification on the moth data, our results show that different sets of learned features can

yield highly variable AUROC scores across different species. Hence, we plan to conduct further experiments with ensemble classifiers constructed from multiple feature sets. Preliminary experiments with one-class classification on the Coprosma data (not shown in this paper), based on features obtained with the generic feature extraction model (InceptionV3 network trained on ImageNet), indicated substantially lower AUROC values across all 17 Coprosma species compared to the results obtained on the moth data and warrant further investigation.

## ACKNOWLEDGMENT

Funded by MBIE Endeavour project 'Biosecure-ID'. We would like to thank Robert Hoare, Jerry Cooper, Brent Martin and David Glenn for their contributions.

## REFERENCES

- [1] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "Cnn features off-the-shelf: An astounding baseline for recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2014.
- [2] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 1717–1724.
- [3] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I*, 2014, pp. 818–833. [Online]. Available: [https://doi.org/10.1007/978-3-319-10590-1\\_53](https://doi.org/10.1007/978-3-319-10590-1_53)
- [4] B. Yao, A. Khosla, and F. Li, "Combining randomization and discrimination for fine-grained image categorization," in *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20-25 June 2011*, 2011, pp. 1577–1584. [Online]. Available: <https://doi.org/10.1109/CVPR.2011.5995368>
- [5] T. Gebru, J. Krause, Y. Wang, D. Chen, J. Deng, and L. Fei-Fei, "Fine-grained car detection for visual census estimation," in *AAAI*, vol. 2, no. 5, 2017, p. 6.
- [6] N. Z. B. C. D. Weihong, J. Hu and J. Guo., "Fine-grained face verification: FGLFW database, baselines, and human-DCMN partnership," pp. 63–73, 2017.
- [7] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," in *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, 2014, pp. 647–655. [Online]. Available: <http://jmlr.org/proceedings/papers/v32/donahue14.html>
- [8] R. Hoare, B. Rhode, and A. Emmerson, "Larger moths of New Zealand: Image gallery and online guide," <http://largemoths.landcareresearch.co.nz/>, 2011, accessed: 20/8/2018.
- [9] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 539–546.
- [10] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *ICML Deep Learning Workshop*, vol. 2, 2015.
- [11] G. Borghi, S. Pini, F. Grazioli, R. Vezzani, and R. Cucchiara, "Face verification from depth using privileged information," in *The 29th British Machine Vision Conference (BMVC)*, 2018.
- [12] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu, "Learning fine-grained image similarity with deep ranking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1386–1393.
- [13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.
- [14] A. Y. Ng and M. I. Jordan, "On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes," in *Advances in neural information processing systems*, 2002, pp. 841–848.