Universiteit
Leiden
The Netherlands

Shape analysis for phenotype characterisation from high-throughput imaging

Guo, Y.; Guo Y.

Citation
Guo, Y. (2017, October 17). *Shape analysis for phenotype characterisation from high-throughput imaging*. *SIKS Dissertation Series*. Retrieved from https://hdl.handle.net/1887/56254

Version:	Not Applicable (or Unknown)
License:	Licence agreement concerning inclusion of doctoral thesis in the Institutional Repository of the University of Leiden
Downloaded from:	https://hdl.handle.net/1887/56254

Note: To cite this publication please use the final published version (if applicable).

Cover Page

Universiteit Leiden

The handle http://hdl.handle.net/1887/56254 holds various files of this Leiden University dissertation

Author: Guo Yuanhao
Title: Shape analysis for phenotype characterisation from high-throughput imaging
Date: 2017-10-17

Chapter 6

Case Study: Image Features and Classification Models

Based on:

- Y. Guo, H. Dibeklioglu & L. van der Maaten, "Graph-based kinship recognition," in IEEE Conference on Pattern Recognition, Stockholm, Sweden, 2014, pp. 4287-4292.
- Y. Guo, C. Liang, F. Lens, R. Vos & F.J. Verbeek, "Image based taxonomy using convolutional neural networks," publication in preparation.

This chapter addresses RQ 6.

RQ 6: *To what extent is it possible that the classification models (or regression models) are able to validate the performance of the image features to characterise the phenotypes in support of shape analysis?*

Abstract – It is difficult to characterise the phenotypes from high-magnification and high-resolution only through the shape analysis. For example, the variation of the local structures of cells and tissues is difficult to represent by the shape description as a whole. Therefore, we propose to, additionally, apply image features to extract the phenotypes encoded in the textures and local structures for the objects in images. Consequently, we use classification models to validate the performance of the applied features on phenotype characterisation. Rather than departing from zebrafish, in this chapter, we use a set of annotated datasets of images, i.e., human faces, a family of butterflies, a family of orchids and an public source for wood species. We aim to develop methods to estimate a structured taxonomy for each of these datasets. For the dataset of human faces, kinship is carefully labelled for pairwise faces and using this dataset, we propose a graphical model to recognise the kinship among a group of people in a family photo (see Section 6.1). In fact, the kinship can be considered as a particular example of taxonomy in which the parents and the children respectively correspond to a parent- and child-node in the hierarchy. For the other datasets, a two-level taxonomy, i.e., the genius and species, are used in the annotations. With the development of feature engineering such the feature learning using a supervised manner, the performance of image classification has been impressively improved. Therefore, we want to investigate representative features for the task of image based taxonomy using the convolutional neural networks (CNN) (see Section 6.2). Experimental results show that our proposed methods have improved the recognition accuracy in both cases. This results in a good understanding of the behaviour of our methods which can be applied in the applications with zebrafish as model system.

6.1 Graphical model for kinship recognition

Genetic correlation among family members is formally represented as kinship, which can be straightforwardly modelled using facial appearance similarity, a particular phenotype. However, due to the diversity of human faces, this phenotype similarity is weak and subsequently presents a challenge to image-based kinship recognition which plays an important role in the application of phenotype characterisation. It is difficult to estimate the kinship from paired faces only through shape analysis. Some prior studies solve the problem of pairwise kinship verification, i.e., on the question of whether two people are kin, through the assessment of the similarity of visual features on images of faces. Such approaches fail to exploit the fact that a global assessment on a group of family members may provide more clues for an accurate kinship recognition; for instance, the probability of two people being brothers increases when both people are recognized to have the same father. In this work, we propose a graphical model that integrates a local kinship confidence, i.e., facial similarity for all pairwise family members in an image, and a global kinship estimation which is represented as a series of reasonable semantic kinship graphs. For a complete and feasible kinship graph, we present an annotated dataset for the kinship of siblings to extend the existing kinship datasets; we also present a dataset of the images with group family members (more than 1) for the performance evaluation of our approach. In our experiments, we have found that the visual features such as Local Binary Patterns can well represent the facial appearance similarity for kinship recognition. The proposed graphical model has improved the accuracy of kinship recognition in group faces.

6.1.1 Kinship recognition using faces

Kinship can be expressed as physiological similarity among family members. For example, parents and children tend to show similar facial appearance and behaviours. In life-sciences, kinship research will support to track genetic evolution of a species. With respect to human beings, facial appearance as an important phenotype can be used as evidence to recognise kinship among different individuals. The image-based kinship recognition has become popular due to its efficiency and reproductivity, which tries to recognise kinship between people based solely

on photographs of their faces. Such application benefits the phenotype characterisation from a large volume of facial images. This may be further helpful in uncovering and analysing social networks, and has applications in surveillance and in criminal investigation. Image-based kinship recognition is a challenging problem: it is a hard task even for humans to recognise kinship among people based on facial similarities. It is encouraging that some recent studies have demonstrated the possibility of kinship verification by means of image-based approaches [128, 129] identifying facial patterns that people may have inherited from their parents. In particular, siblings have the same gene sources which results in the presence of similar facial features. Facial cues that are informative for kinship recognition include the colour and shape of the eyes, eyebrows, nose, and mouth [130].

Prior work on image-based kinship recognition has three main limitations. First, prior studies only consider kinship verification: they try to determine whether kinship exists between a pair of faces, but they do not aim at recognising the exact type of kinship [129, 131, 132, 133]. Second, current kinship datasets are insufficient for the evaluation of existing kinship recognition algorithms, in particular, because existing datasets do not contain examples of siblings. Third, prior studies only consider settings in which kinship needs to be verified between pairs of people. This does not correspond to the typical setting encountered on social network websites, on which people often upload photographs that contain more than two family members. One may deal with this problem by separately classifying all pairs of faces in the family picture, but such an approach fails to share information between the pairs of people and may produce classifications that are inconsistent (*e.g.*, two people may be classified as sisters whilst they are also classified as having different parents).

Motivated by the aforementioned problems of prior work in kinship recognition, we study image-based kinship recognition in photographs that contain several family members. Specifically, this section makes three main contributions. First, we focus on kinship *recognition* instead of kinship *verification*: we aim to recognise the type of kinship relations between people. Second, we introduce two new datasets: (a) an annotated dataset containing photographs of siblings and (b) an annotated dataset of family photographs. The latter dataset and part of the former dataset is made publicly available. Third, we propose a novel graph-based algorithm that performs joint kinship recognition of all faces in a family picture.

The general framework of this algorithm is illustrated in Figure 6.1. The key advantage of our graph-based algorithm is that it exploits the fact that in a normal family, the recognised kinship of a particular pair of faces provides evidence for (non)kinship between other pairs of people. For example, in a family, two siblings should have the same father and mother[1]: if A and B are brothers and C is the father of A, then C must also be the father of B. Our graph-based algorithm constructs a fully connected graph in which faces are represented by vertices and kinship relations between pairs of faces are represented as edges. Using a few simple kinship rules (that are shown in Table 6.1), we can generate all valid kinship graphs. For each new test image, the predicted kinship graph is the one that obtains the highest score when we sum all scores of the pairwise classifiers that correspond to the edges. Because our graph-based algorithm shares information between the pairwise classifiers, ambiguities in the pairwise kinship classifications may be resolved, which may lead to improved performance. The results of our experiments demonstrate that the proposed algorithm can substantially improve kinship recognition accuracy.

6.1.2 Previous work

Most prior studies on image-based kinship recognition aim to solve the kinship verification problem using computer vision and machine learning techniques [129, 131, 132, 133]. All these approaches extract facial features and train a kinship verification classifier on a collection of annotated examples. In the seminal paper on automatic kinship detection [129], facial resemblance is represented by the difference between facial features. The extracted features include face colour, the position and shape of face parts, as well as gradient histograms. Face parts are localised using a pictorial structures model [134]. Classification is performed using a k-nearest neighbour classifier. [129] presents experiments in which the performance of an automatic kinship verification system is compared with human performance; the results show that the proposed algorithm performs 4.9% better than human accuracy on this task. [132] improves over this method by dropping the assumption that kinship examples have higher feature similarities than non-kinship examples. They learn a distance metric that aims to repel non-kinship samples as far as possible, whilst kinship samples are pulled close. The method of

[1] In this study, step relationships are not considered.

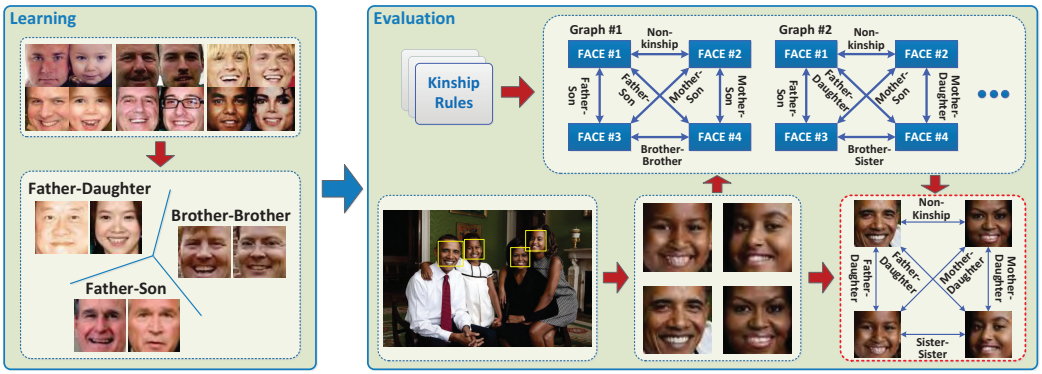


Figure 6.1: Overview of the proposed kinship recognition system. In the learning phase, a multi-class kinship classifier is jointly trained on different kinship relations. In the evaluation phase, the faces in family photographs are detected, cropped, and normalised. The set of all valid kinship graphs is generated according to the constraints on kinship relations. For each resulting candidate graph, the classifier scores are summed to obtain an overall score. The kinship graph with the highest overall score is selected as the prediction.

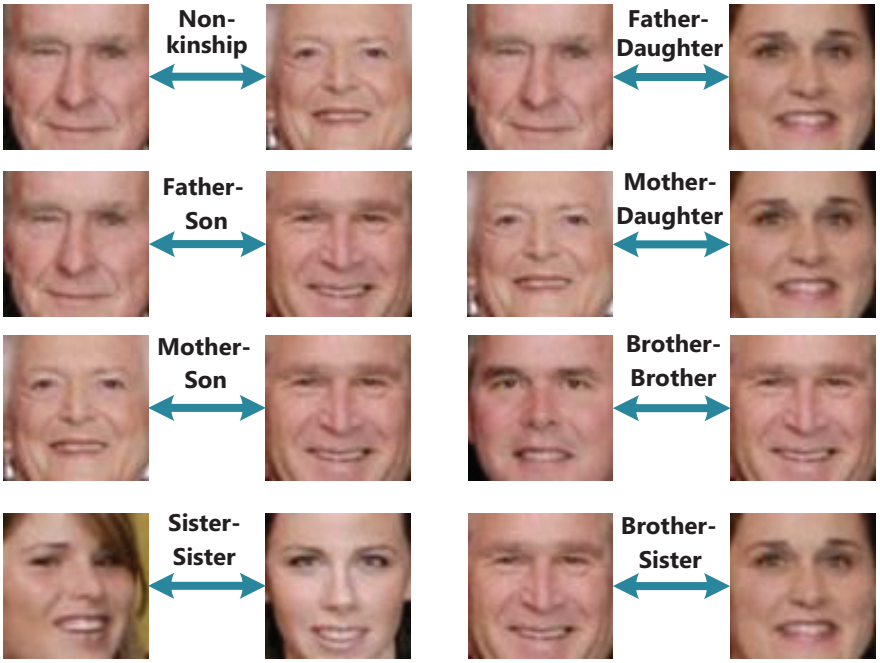


Figure 6.2: Normalised face pairs (from the Group-Face dataset) showing different kinship relations.

[132] also combines different types of feature descriptors by learning a multiview distance metric.

In [131] and [135], Xia *et al.* propose to use transfer subspace learning methods for kinship verification. They exploit the idea that the kinship verification between children and their parents is easier when the parents are young. The method learns a subspace in which old parents and their children are projected close together; the subspace model can then be used to make images of parents look younger. Recently, Dibeklioğlu *et al.* have proposed a method that uses facial expression dynamics combined with spatio-temporal appearance features to verify kinship in videos [133]. This method is based on the observation that the dynamics of facial expressions are informative for kinship recognition based on videos of people.

In contrast to the aforementioned methods, [136] does not focus on kinship verification but aims at recognising whether a group picture is a family picture. The method estimates the gender and age of every face in the group picture. An image graph is constructed by fitting a minimum spanning tree based on the face locations. Subsequently, the image is represented as a bag of image subgraphs. The resulting bag-of-image-subgraph features are then used to determine whether the group picture is a family picture. The method, however, does not recognise the types of kinship that are present within the family picture.

Our work has several differences in comparison to prior studies. First of all, instead of verifying kin relationships, our study focusses on recognising the exact type of kinship relations. Additionally, our study is the first attempt to generate complete kinship graphs for family photographs.

6.1.3 Graphical model for kinship recognition

Here, we propose an automatic kinship recognition system that relies on graph-based optimization of multi-class kinship classification. This work does not consider kinship verification between face pairs but focusses on classifying the type of kin relations. Assuming that kin pairs are known in a given group photograph (or predicted by an existing kinship verification system), our system predicts a kinship graph that describes the kinship relations between the family members.

(A) Feature extraction

Table 6.1: Kinship graph generation rules

Definition	Instance
• One child can at most have one father and one mother.	(A-B:Father-Daughter/Son) $\Rightarrow \neg$ (C-B:Father-Daughter/Son) (A-B:Mother-Daughter/Son) $\Rightarrow \neg$ (C-B:Mother-Daughter/Son)
• Siblings have the same parents.	[(A-B:Father/Mother-Daughter/Son) $\wedge$ (A-C:Father/Mother-Daughter/Son)] $\Rightarrow$ (B-C:Sister/Brother-Sister/Brother)
• Siblings have the same siblings.	[(A-B:Sister/Brother-Sister/Brother) $\wedge$ (A-C:Sister/Brother-Sister/Brother)] $\Rightarrow$ (B-C:Sister/Brother-Sister/Brother)
• There should not be kinship between father and mother.	[(A-B:Father-Daughter/Son) $\wedge$ (C-B:Mother-Daughter/Son)] $\Rightarrow$ (A-C:Non-kinship)

For the reliability of similarity analysis, face images need to be aligned before the feature extraction step. To this end, eye corners are located using the facial landmarking method proposed in [137]. Based on the eye locations, faces are aligned (in terms of roll rotation, translation, and scale) and cropped. The size of the resulting images are 64×64 pixels. Figure 6.2 shows samples of the normalised faces.

To describe the facial appearance, we use Local Binary Pattern (LBP) features [138]. Following [133], LBP features are extracted from each cell in a 7×5 grid that is imposed over the normalised face. In addition to LBP appearance features, we also extract gender and age features from the face images.

In order to estimate a gender feature $f_{\text{gender}}(I_i) \in \{-1, +1\}$ for a given face image I_i, we classify LBP and bio-inspired features (BIF) [139] using a binary support vector machine (SVM) classifier (with radial basis function kernel). Additionally, we extract an age feature $f_{\text{age}}(I_i, I_j) \in \{-1, 0, +1\}$ that describes the relative age of the given face images I_i and I_j:

$$f_{\text{age}}(I_i, I_j) = \begin{cases} -1: & a(I_i) < a(I_j) \\ 0: & a(I_i) \cong a(I_j) \\ +1: & a(I_i) > a(I_j) \end{cases}, \tag{6.1}$$

where a denotes the true age of the given subject. For the estimation of f_{age}, we employ a three-class SVM classifier using BIF features. To obtain the final feature vector for a pair of face images (I_i, I_j), all features are concatenated:

$$\mathbf{x}_{ij} = [f_{\text{LBP}}(I_i), f_{\text{LBP}}(I_j), f_{\text{gender}}(I_i), f_{\text{gender}}(I_j), f_{\text{age}}(I_i, I_j)] \,.$$

(B) Pairwise kinship classification

We model the resulting feature vectors to be able to distinguish between different kinship types. Moreover, we aim to predict the direction of these relations. For instance, the estimation for the given images will be that I_i is the father of son I_j (father→son), instead of just indicating that I_i and I_j have father-son relation. To this end, we define 12 types of directional kinship relations such as father→daughter, father←daughter, father→son, father←son, mother→daughter, mother←daughter, mother→son, mother←son, brother→sister, brother←sister, brother-brother, and sister-sister. By using these kinship types, more distant kinship relationships such as grandparents↔grandchildren, cousins, and uncle/aunt-nephew/niece may also be inferred if the family picture also contains the "intermediate" people.

We use a multi-class linear logistic regressor (LR) as the classifier in our system. For a pair of face images, the predicted label $\mathbf{y}^*$ is thus given by:

$$\mathbf{y}^* = \underset{\mathbf{y}}{\operatorname{argmax}}\ \mathbf{y}^\top \left(\mathbf{W}^\top \mathbf{x} + \mathbf{b}\right), \tag{6.2}$$

where $\mathbf{y}$ is a 1-of-K label vector. $\mathbf{W}$ and $\mathbf{b}$ denote the classifier weights and bias, respectively. To train the multi-class logistic regressor, we define the class-conditional probability:

$$p(\mathbf{y}|\mathbf{x}) = \frac{\exp(\mathbf{y}^\top(\mathbf{W}^\top \mathbf{x} + \mathbf{b}))}{\sum_{\mathbf{y}'} \exp(\mathbf{y}'^\top(\mathbf{W}^\top \mathbf{x} + \mathbf{b}))}. \tag{6.3}$$

In our application, this probability represents the likelihood of the kinship type given a pair of faces. We aim to minimize the penalized conditional log-likelihood $\mathcal{L}$:

$$\mathcal{L}(\mathbf{W}, \mathbf{b}) = \underset{\mathbf{W}}{\operatorname{argmax}} \left(\sum_{\mathbf{x}} \log p(\mathbf{y}|\mathbf{x}) - \lambda \|\mathbf{W}\|_2^2 \right). \tag{6.4}$$

Herein, the second term is an L2-norm regulariser that is employed to prevent overfitting. The value of the regularisation λ is set based on the error measured on a small, held-out validation set.

(C) Kinship graphs

A straightforward way to recognise kinship relations in a family photograph is to classify each pair of faces individually. However, this approach does not share information between the pairwise classifications: if the classifier doubts between

two kinship types, individual classification cannot exploit the other kinship relations in the photo to resolve this ambiguity. Individual classification may even produce infeasible kinship graphs. For example, it may predict that two people are brothers whilst predicting that they have different parents. The graph-based algorithm we propose aims to resolve these two problems by: (1) generating all feasible kinship graphs and (2) selecting the kinship graph that obtains the highest score.

A kinship graph can be defined as $G = (V, E)$ in which faces correspond to vertices and edges to kinship relations. In other words, each edge $(i, j) \in E$ has an associated label $\mathbf{y}_{ij}$. Two examples of kinship graphs using three faces are shown in Figure 6.3. Note that the graph shown in Figure 6.3(b) is actually infeasible since it violates the constraints on kinship relations that are given in Table 6.1. In the first step, all possible kinship graphs that satisfy these constraints are generated. It is important to note that the candidate graphs can actually be generated offline. The resulting set of candidate kinship graphs are denoted by $\mathscr{G}$. Afterwards, we assign a score to each of the candidate kinship graphs that measures the (log)likelihood of that kinship graph for the observed family picture. Specifically, we define the kinship graph score as the sum of the kinship classifier scores that correspond to each of the edges in the graph:

$$s(G|\mathcal{I}) = \sum_{(i,j)\in E} \mathbf{y}_{ij}^{\top} \left(\mathbf{W}^{\top}\mathbf{x}_{ij} + \mathbf{b}\right), \tag{6.5}$$

where $\mathcal{I}$ is the family photo, $G = (V, E)$ is the kinship graph that we are scoring, $\mathbf{x}_{ij}$ is the feature vector extracted from the pair of faces associated to edge $(i, j) \in E$, and $\mathbf{y}_{ij}$ is the corresponding kinship label. We perform kinship graph prediction for family photo $\mathcal{I}$ by maximising the graph score over the set of all candidate kinship graphs:

$$G^{*} = \underset{G\in\mathscr{G}}{\operatorname{argmax}}\ s(G|\mathcal{I}), \tag{6.6}$$

where graph G^{*} is the predicted kinship graph.

6.1.4 New datasets

To evaluate our approach, we gathered two new kinship recognition datasets: (A) a dataset with image pairs of siblings and (B) a dataset with family photographs.

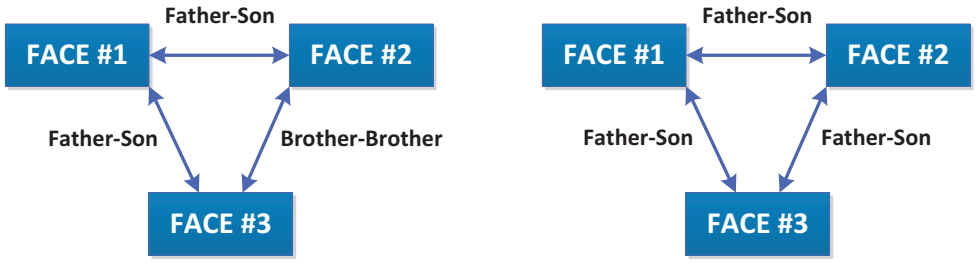


Figure 6.3: Samples of (a) feasible and (b) infeasible kinship graphs.

Figure 6.4: Sample images from the Group-Face dataset.

Both datasets (except some copyrighted images in the first dataset) are made available to the research community. One can find the dataset at `https://pan.baidu.com/s/1nvPxQ8D` (pincode: e8if). Both datasets are described separately below.

(A) Sibling-Face dataset

Existing large-scale kinship datasets (such as the KFW-II dataset [132]) do not

Table 6.2: Distribution of kin pairs (image pairs) in the KFW-II, Sibling-Face and Group-Face datasets.

	KFW-II	Sibling-Face	Group-Face
Father-Daughter	250	-	69
Father-Son	250	-	69
Mother-Daughter	250	-	70
Mother-Son	250	-	62
Brother-Brother	-	232	40
Sister-Sister	-	211	32
Brother-Sister	-	277	53

include sibling pairs. The UvA-NEMO dataset [133, 140] contains sibling pairs, but it has a small number of subjects. We have gathered a new dataset that contains more than 200 image pairs for each of three possible sibling relations (brother-brother, sister-sister, and brother-sister). All sibling images have been collected from websites such as Flickr; the sibling relations have been determined based on the tags or descriptions of the images. The sibling faces have been processed in the same way as done for the images in the KFW-II dataset: they are aligned according to the position of eyes, and resized to a fixed size of 64×64 pixels. In our experiments, the Sibling-Face dataset is combined with the KFW-II dataset to train kinship classifiers. The distribution of kin pairs in the KFW-II and Sibling-Face datasets is given in Table 6.2.

(B) Group-Face dataset

We have also gathered a collection of group photographs from publicly available sources such as Flickr. Specifically, we have selected group pictures in which the people are all frontally facing the camera. Some samples from the collected dataset are shown in Figure 6.4. The dataset consists of 106 group photographs, of which 82 contain group(s) of family members. To facilitate labelling of the kinship relations, we have selected photographs of famous families (royalty, presidents, Hollywood stars, etc.) and photographs of regular families with reliable kin labels. The Group-Face dataset contains father-daughter (FD), father-son (FS), mother-daughter (MD), mother-son (MS), brother-brother (BB), sister-sister (SS) and brother-sister (BS) pairs. Table 6.2 shows the number of image pairs in each kinship class. All the faces in the dataset have been cropped and aligned in the same way as the faces in the Sibling-Face dataset.

6.1.5 Experimental Results

In our experiments, the KFW-II and Sibling-Face datasets are combined and used for training. We employ the family photos in our Group-Face dataset as the test set. It is assumed that we know which pairs of faces in the family pictures have kinship and which pairs of faces do not, i.e., we assume that we have access to a perfect kinship verification algorithm and focus solely on recognising what type of kinship exists between two people. In our experiments, the maximum number of family members is limited to four because, in our current (naive) implementation, the total number of candidate kinship graphs and the required amount of memory drastically increases when more than four faces are used. Specifically, when a family photo contains two parents and four children, we manually split the family into two groups which both have parents and two children. In this way, we obtained 98 kinship groups (16 groups with two faces, 40 groups with three faces, and 42 groups with four faces) that we use in our kinship recognition experiments. The test set we used in our experiments is made publicly available (as part of the Group-Face dataset).

As a baseline approach, we individually perform pairwise classification on each edge of the kinship graph to determine the type of kinship. We set the regularisation parameter λ (see Equation 6.4) of the kinship classifier by cross-validating over a small held-out validation set.

To test the reliability and efficacy of the proposed graph-based kinship recognition, we perform two different experiments. In the first experiment, kinship recognition performances of the graph-based and pairwise approaches are compared. In the second experiment, we investigate the effect of age/gender estimation accuracy on the robustness of the graph-based and pairwise methods. To this end, we systematically perturb the gender and age features which are extracted from the test data. The details and results of these experiments are given below.

(A) Graph-based versus pairwise classification

In this experiment, the correct classification rates of the graph-based and pairwise approaches are compared. As shown in Table 6.3, the graph-based method proposed in our study outperforms the pairwise kinship classification by 16.77% (absolute) on average. This result demonstrates the efficacy of the graph-based kinship recognition. The highest performance of the graph-based method is achieved for the sister-sister relationship with an accuracy of 76.92%.

Table 6.3: Kinship recognition accuracy of the pairwise (baseline) and the graph-based approaches.

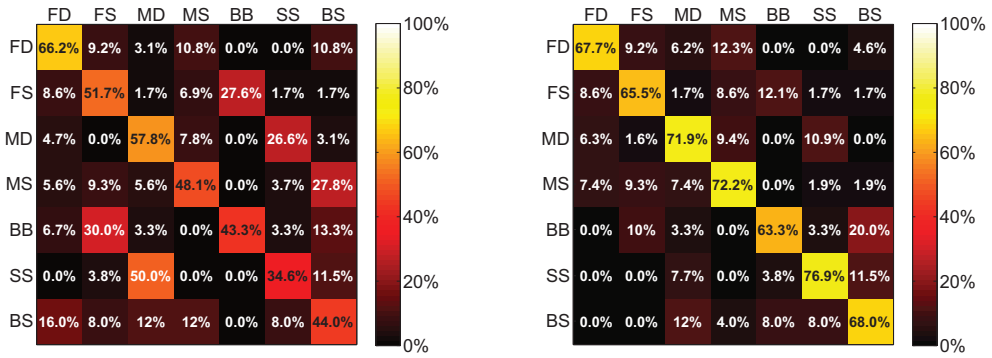
Relationship	Pairwise (%)	Graph-based (%)	# Test Pairs
Father-Daughter	66.15	67.69	65
Father-Son	51.72	65.52	58
Mother-Daughter	57.81	71.88	64
Mother-Son	48.15	72.22	54
Brother-Brother	43.33	63.33	30
Sister-Sister	34.62	76.92	26
Brother-Sister	44.00	68.00	25
All	52.48	69.25	322

For further exploration of the results, the confusion matrices for both methods are given in Figure 6.5. The results suggest that, unlike the pairwise classification, the graph-based approach is able to recover from errors in the age/gender estimations. For instance, the baseline approach often confuses the father-son relation with the brother-brother relation, presumably due to errors in the relative age estimation[1]. By contrast, the graph-based approach corrects most of such misclassifications by incorporating other relations in the graph, and by ensuring that the predicted kinship graph is feasible. This is confirmed by the number of kinship graphs which are correctly predicted (*completely*) on the Group-Face dataset. Whilst the graph-based approach correctly predicts 56 of 98 kinship graphs, only 29 kinship graphs are correctly recognised by the baseline method.

(B) Effect of age and gender estimation accuracy

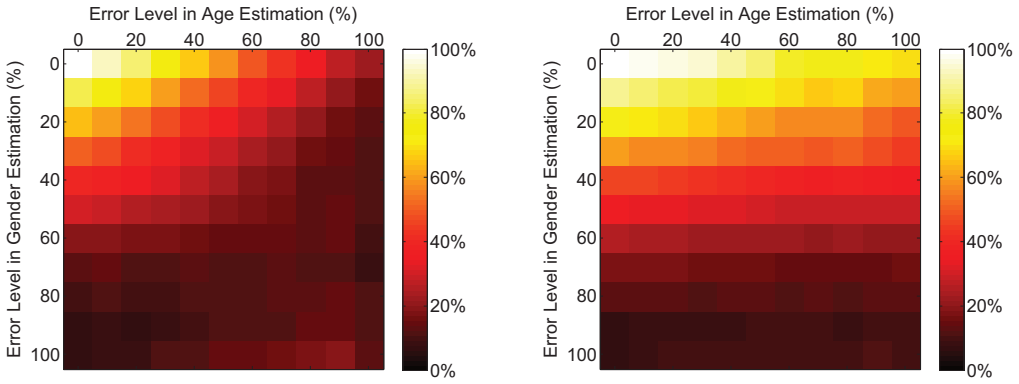
The results presented in the previous subsection illustrate the potential merits of our graph-based algorithm, which mainly stem from its ability to correct errors in the age and gender estimations. We further investigate the effect of age and gender estimation accuracy in our method. To this end, we randomly generate labels for the relative age classes and gender by systematically changing the error rate. Both the graph-based and pairwise methods are tested using these labels.

[1]The correct classification rate of the gender classifier, used in our experiments, is approximately 90% based on 10-fold cross-validation. Combination of the KFW-II, Sibling-Face, and UvA-NEMO datasets is used for the evaluation. 10-fold cross-validation accuracy of the relative age estimator is approximately 65% on the combination of KFW-II and Sibling-Face datasets. Higher error rate in age estimation is mostly due to small size (low resolution) of the face images, which makes facial wrinkles nearly invisible.



(a) (b)

Figure 6.5: Confusion matrices for (a) the pairwise and (b) graph-based approaches.



(a) (b)

Figure 6.6: Kinship recognition accuracy (%) as a function of the error level in age and gender estimation for (a) pairwise and (b) graphical model of kinship recognition.

Figure 6.6 shows the kinship recognition accuracy as a function of the error level in age and gender estimation. As shown in Figure 6.6, both methods achieve 100% classification accuracy when the age and gender ground truths are used: age and gender completely determine the type of kinship relation between two people, if we assume that the given pair has kinship.

The results show that both pairwise and graph-based approaches perform worse when the perturbation rate is increased for gender and age. However, our graph-based method is more robust to gender and age estimation errors than to the pairwise approach. In particular, the graph-based algorithm is less sensitive to

incorrect age prediction. This is beneficial because age estimation is a difficult task in real-life conditions, in particular, because age estimates are strongly influenced by changes in resolution, illumination, gender [141], and facial expression [142]. Our graph-based algorithm is more robust to the resulting errors in the age estimates. As shown in Figure 6.6 (see top right side of the accuracy maps), graph-based approach performs much better than the pairwise classifier in such conditions.

6.1.6 Section conclusions and future work

In this section, we have proposed a novel graph-based method to recognise kinship relations in family photos. It partially answers RQ 6: *To what extent is it possible that the classification models (or regression models) are able to validate the performance of the image features to characterise the phenotypes in support of shape analysis*? Our approach models the kin relationships using a fully connected graph in which faces are represented by vertices and edges represent kinship relations. The overall score of each feasible kinship graph is computed by summing classifier scores over the edges of the graph. The graph with the highest overall score is selected as the prediction. The results of our experiments demonstrate that our graph-based outperforms the pairwise kinship classification approach. Moreover, the proposed method guarantees consistency of the predicted kinship graphs.

We consider that RQ 6 is partially answered that the graphical model and a classification model, i.e., the logistic regression, have cooperated to validate the performance of the LBP features in the application of image based kinship recognition. It turns out that the well-designed image features will be able to characterise the subtle variation of the phenotypes such as shape and texture.

As a future direction, we aim to develop a graph-based method to train our kinship classifier as well by framing the task as a structured prediction problem. Also, we aim to improve the speed of our current (naive) implementation by exploiting redundancies in the score computations (like in dynamic programming). Moreover, we plan to include a kinship verification step prior to the classification of relations. Finally, we will apply the method in the applications which use the zebrafish as model system.

6.2 Image based taxonomy using CNN

Phenotypes including shape and texture represented in appearance are essential in image based taxonomic classification of biological specimens. This presents a challenge to the choice of features to generalise these phenotypes. We are motivated to investigate representative features for the task of image based taxonomy using the convolutional neural networks (CNN). We first present three dataset with a taxonomic structure, which include orchids, butterflies as well as introduce an open source for wood species (in fact, the kinship addressed in Section 6.1 is a special category of the taxonomic structure). We adapt a popular CNN architecture, the VGGNet-16, to learn representative features for these tasks in a supervised manner. We implement a multi-output layer of which each output corresponds to a flat classifier for each level in the taxonomy. In this manner, we can introduce multi-supervision to the training time of the networks. This avoids to learn individual classifiers on each level or each node which is commonly used in conventional hierarchical classification. We use a fine-tuning strategy to accelerate and stabilise the training process. Experimental results show that the proposed approach achieves better performance compared to the methods using hand-crafted features and pre-trained networks. From our observation, representative features are of great importance to a well-performing recognition system for taxonomy. Importantly, in our method the prediction for each level in the taxonomy can be performed in one forward pass.

6.2.1 Image based taxonomy

A feasible and convenient manner for categorisation gives rise to digitization, reuse and efficient management for the large amount of the collection of cultural heritage. Under these circumstances, taxonomic categories are commonly used, which formally use a hierarchical ranking i.e., Kingdom, Phylum, Class, Order, Family, Genus and Species to categorise and annotate the specimens [1, 143]. This manner also facilitates an efficient top-to-bottom data retrieval. In practice, a taxonomic recognition system will also facilitate many applications such as recognition of endangered species [144].

Using imaging of specimens makes image based taxonomy possible. It aims to learn a model to recognise each rank in the taxonomy for a specimen using images which represent that specimen as a whole or microscopic structure. In practice,

researchers in life-sciences make use of their expertise to identify the species of a specimen [145]. However, some species, for example, the ones in the same genus, present rather similar shapes; subsequently, their textures such as special patterns on the specimen surface should be emphasised. Therefore, the image based taxonomy requires comprehensive investigation of phenotypes including shape and texture in the whole appearance of the specimen.

Image based taxonomy is, in fact, a typical hierarchical classification problem [146]. Each level in the hierarchy represents a rank in the taxonomy. In a task of image based taxonomy, it is usually easy to recognise a higher rank due to the remarkable dissimilarity of appearance for the specimens from different classes; and it is usually difficult to recognise a lower rank due to the dramatic similarity of appearance for the specimens of which the classes share the same parent rank. Therefore, a proper choice of feature representation for phenotypes can result in a well-performed taxonomic recognition system. For example, experts can accurately recognise a wood species through a careful investigation on microscopic features such as shape and size of vessels and fibrous structure of tissues [147].

In practice, we have multiple options of image features. For the last decades, many local features have been increasingly used for image recognition, such as Histograms of Oriented Gradients (HOG) [24], Local Binary Patterns (LBP) [138] and Scale Invariant Feature Transform (SIFT) [109]. There are also many available shape features, such as shape context [148], the angular radial transform [149] and projective invariant contexts [150]. Some of these features are generic and suitable to the problem of image based taxonomy; some are well-designed for a particular domain. With the fast development of deep convolutional neural networks (CNN) [25, 151, 152, 153], successful applications have been made in many fields like computer vision [64, 154, 155] and gaming [156]. The readout of a deep CNN architecture is, in fact, a feature engineering which learns discriminative features from images in a supervised manner. This makes the deep CNN architecture very flexible for learning representative features for corresponding applications. The current development has inspired us to apply the CNN architecture in the image based taxonomy due to the diversity of taxonomic categorisation as for each specific taxonomic category, different features should be emphasised.

Here, we first present three taxonomic structured datasets with expert tags; Javanese butterflies, slipper orchids and wood species. The first one is obtained from a collection of Dutch National Natural History Museum (Naturalis Biodiversity Center `http://www.naturalis.nl/`) for the family of Papilionidae. The second one is obtained from some public sources such as ImageNet [157] for the family of Cypripedioideae. Both datasets are labelled by a two level taxonomy: genus and species. The third dataset is a public source for microscope images of wood species [158]. This dataset contains a more specific taxonomic structure from class to species. In this work, we only employ two level annotations including class and species.

We are motivated to present a CNN architecture based on the VGGNet [151] which is extended with a multi-output layer for the image based taxonomy. Each output corresponds to a local classifier for each level in the taxonomy. We train the whole networks considering all the taxonomic annotations for each example to be trained. This means that the multi-supervision jointly contributes to the training phase. We use a fine-tuning strategy for the training of the networks. We first introduce a pre-trained model using a large dataset such as ImageNet and then use our datasets to enhance the representability of the networks for our application. This operation largely stabilises and accelerates the training of the network.

In an hierarchical classification, the proposed method can be categorised as a local classification per level approach, which is also referred to as top-down strategy [159]. This method may introduce the problem of label inconsistency. For instance, a testing example may be assigned labels that not refer to a reasonable parent-child routine. This procedure can be improved by post processing. Another possible solution is to use the flat classification approach. Such an approach only trains a classifier for the bottom level and a bottom-top strategy can be used to back-propagate the labels on higher level according the deterministic property of the parent-child mode [160]. Other attempts concern global classification models [161, 162, 163, 164]. We should note that all these methods mainly focus on a classification model to generalise the hierarchical classification problem. The method can be considered to improve the output layer in our application of image based taxonomy. Here, we would like to focus on the contribution of features in our particular problem. Therefore, we first use a simple classification strategy e.g., softmax [33], as a local classifier for each level to validate the performance of our representative features.

Actually, the hierarchical classification problem can be considered as a special case of multi-label classification [165]. Many deep CNN architectures have been reported to solve this problem [166, 167, 168]. Recently, a hierarchical deep CNN (HD-CNN) architecture has been reported, which presented a coarse-to-fine strategy for a large scale of visual recognition [169]. This enables the so-called local classifier per node approach with a CNN architecture. Here, we stress the importance of features in our image based taxonomy of biological specimens. So, we propose to extend the CNN architecture with a multi-output layer, of which each output corresponds to a level in the taxonomy. This will provide a good understanding of the performance of the features for each level. In future work, we can consider to introduce a dedicated architecture such as the HD-CNN in our problem.

6.2.2 Image based taxonomy using CNN architecture

We first present (A) the datasets used in this work, and (B) elaborate in details the CNN architecture we have adapted.

(A) Datasets

Below we briefly discuss the datasets, i.e., Butterflies, Orchids and Woods.

The dataset of *Butterflies* are obtained from a large collection for the family of Papilionidae, a category of Javanese butterflies caught in the 1930s. With the development of digitalisation of cultural heritage, images have been made for these specimens. Some examples can be seen in Fig. 6.7 (A). In the images, the specimens are well-positioned on their profile-view and most of the features such as the texture and patterns on their wings are clearly presented. In this manner, we avoid the effects of shape misalignment and scaling. This dataset is structured in a two taxonomic categories, i.e., genus and species. Until now, the dataset consists of 1829 images which are from 18 genera and 45 species.

The dataset of *Orchids* are obtained for the family of Cypripedioidea. The orchid experts annotated 1117 images with 5 genera and 116 species [170]. Examples can be seen in Fig. 6.7 (B). One should note that the datasets of *Butterflies* and *Orchids* have the problem of data imbalance. Some classes only contains a small number of examples and some others contain much more. This will present a challenge to a classification model which may result in overfitting for the classes with a large number of examples.

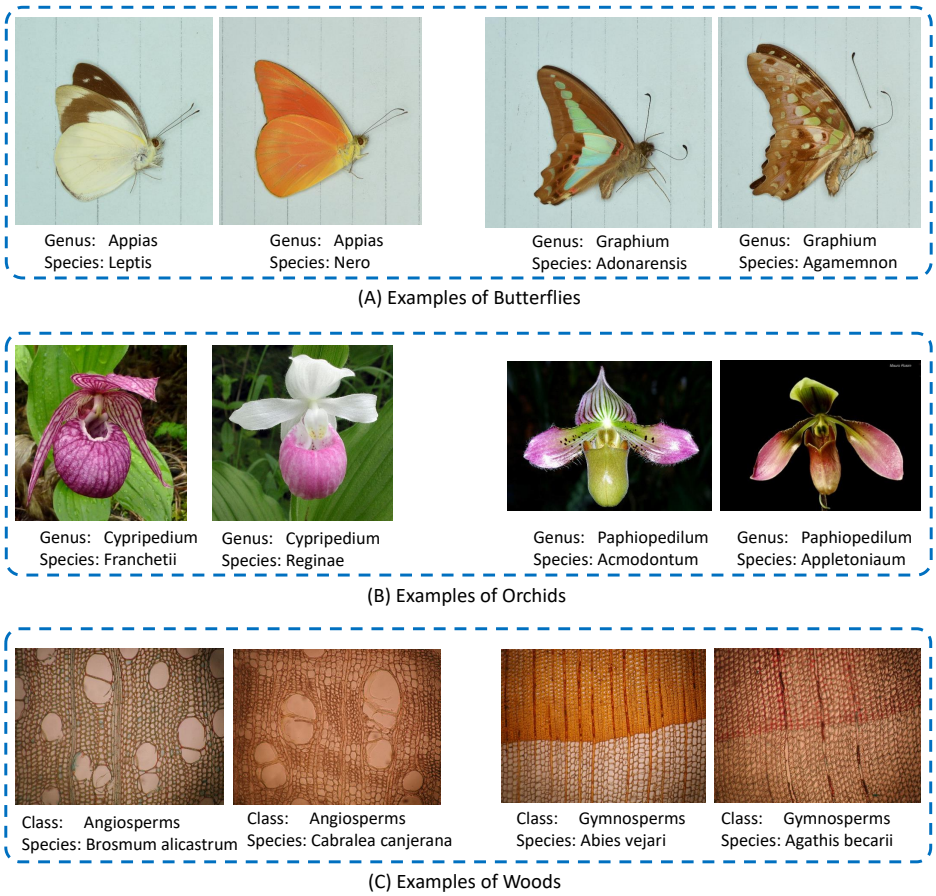


Figure 6.7: Examples of the images from dataset (A) Butterflies (B) Orchids and (C) Woods. For each dataset, we select four examples from four species, two of which are from the same genus (class). One can observe that, the phenotypes, such as the colour and patterns on butterfly's wings, the shape and the texture of orchid's pedals, the shape and structure of wood's vessel and tissue, show significant similarity from the species which share the same genus (class).

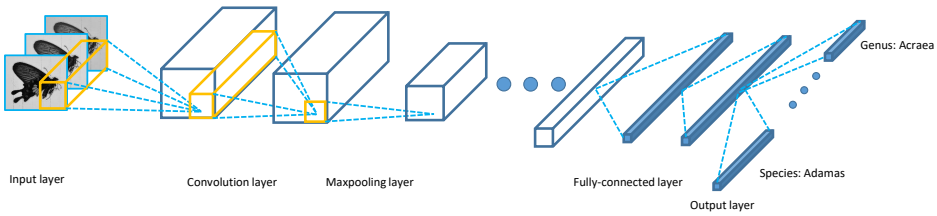


Figure 6.8: The CNN architecture with a multi-output layer. Only one filter is shown for the convolution and pooling layers.

The dataset of *Woods* was originally presented in [158]. It contains 2240 wood images from 2 classes and 112 species. This dataset has an even distribution as each species contains 20 examples. The images from stained wood slices are acquired using a microscope (Olympus Cx40) with a 100x objective. The size of the acquired RGB images is 1024x768. Examples of the images can be found in Fig. 6.7 (C).

From the datasets, we can see a remarkable diversity of the task to design the image based taxonomy. It is difficult to design a type of generalised feature for this application with diverse objectives. For example, the HOG features works fine in the butterflies and orchids but totally fails in the woods. This requires a generalised framework for feature engineering to obtain more discriminative and representative features for each task. We can also observe that the datasets of *Butterflies* and *Woods* are produced in specific imaging conditions. The acquired images are all standard, for example, the orientation and position of the specimen represented in the images is stable. The dataset of *Orchids* is more challenging because the examples are collected in a natural setting.

(B) VGGNet with a multi-output layer

A standard CNN architecture consists of one input layer, a set of convolution layers, several pooling layers, one or two fully-connected layers and one or multiple outputs layer. In Fig. 6.8, we show a schematic representation of the CNN architecture with a multi-output layer.

The input layer is also referred to as data layer which converts the input image into the format a CNN architecture requires. The convolution layer generates a set of feature maps through convolving the previous feature maps using different filters. The weights of a filter are shared by the whole convolution which produces one feature map. This means each element in a feature map corresponds to a receptive field from the original image. One should note that a non-linear operation of rectification such as ReLU [25] is performed after each convolution layer. The pooling layer aims to subsample a feature map, to an extent holding good spatial property in the feature representation. Similar to the conventional multi-layer perceptron [171], the fully-connected layer connects all the elements in previous layer to each of the neurons in the fully-connected layer. This operation converts the feature map into a one-dimensional feature vector. The output layer can be a fully-connected layer which can be followed by a loss in training time. The supervision of the network training is implemented in this process. In this

supervised manner, the parameters in a CNN architecture can be obtained using a standard algorithm such as gradient back-propagation. As a result, a well-trained CNN architecture can largely fit the training data and the extracted feature maps can be discriminative and representative to our task, i.e., image based taxonomic recognition.

We have adapted the VGGNet-16 in our CNN architecture. In order to produce taxonomic categories, we adapted the last layer of VGGNet-16, i.e., the output layer, with a multi-output layer. Each output corresponds to a level in the taxonomy. The networks consists of 14 convolutional layers, 5 pooling layers, 2 fully-connected layers and 1 multi-output layer. More concrete, the configuration is depicted as follows: Input image ($224 \times 224 \times 3$) $\rightarrow$ 2 (3×3) convolution layers (64 feature maps) $\rightarrow$ maxpooling layer $\rightarrow$ 2 (3×3) convolution layers (128 feature maps) $\rightarrow$ maxpooling layer $\rightarrow$ 3 (3×3) convolution layers (256 feature maps) $\rightarrow$ maxpooling layer $\rightarrow$ 3 (3×3) convolution layers (512 feature maps) $\rightarrow$ maxpooling layer $\rightarrow$ 3 (3×3) convolution layers (512 feature maps) $\rightarrow$ maxpooling layer $\rightarrow$ fully-connected layer (4096) $\rightarrow$ fully-connected layer (4096) $\rightarrow$ multi-output layer (softmax).

We use the library of Caffe [172] in our implementation. Both for the training and testing, we re-scale all the images into a size of 256×256 pixels. At training time, we use a pre-trained model to initialise the weight layers including all the convolution layers and the 2 fully-connected layers. For the last fully-connected layer, i.e., the multi-output layer, we initialise the weights using a Gaussian distribution with the mean as 0 and the standard deviation as 0.01. We use the statistical gradient descent strategy to train the networks and we set the batch size as 64. We set the total iterations as 2000 and the learning rate as 5×10^{-3}. We decay the learning rate as half of the original value after 1000 iterations. We train and test our model using two NVIDIA TITAN X GPUs.

6.2.3 Experiments

In this subsection, we apply our CNN architecture on the datasets to evaluate the its performance in the task of image based taxonomy. (A) We compare the performance by different methods. (B) We discuss the classification results using confusion matrix and visualise the representative features from our CNN architecture using the t-SNE map [173].

(A) Performance evaluation with different features

In this experiment, we perform cross validation on the datasets using different methods. Due to the data imbalance in the datasets of *Butterflies* and *Orchids*, we leave out the classes with less than 3 examples and finally we use 3-fold cross validation. For the dataset of *Woods*, we use 5-fold cross validation. We randomize the partition of the folds and repeat the whole process for 5 times to obtain a statistical representation for the accuracy. In Table 6.4, we separately report the accuracy for the two levels in the taxonomy of the datasets. In each row of Table 6.4, the upper value corresponds to the accuracy for the prediction of genus (class), and the lower value represents the accuracy for the prediction of species.

A1. Configuration We use two popular features, the rotation-invariant uniform LBP and HOG, as comparisons in this experiment. In order to obtain the identical feature dimensions in each dataset, we rescale the images from *Butterflies* and *Orchids* to 256×256 pixels; and we keep the original image size for the *Woods*, i.e., 1024×768.

For the LBP, we configure the sampling radius and the number of sampling points as (2,8) for the *Butterflies* and *Orchids*; (3,24) for the dataset of *Woods* due to its large image size. The former results in a 59-dimensional feature vector; the latter produces a 555-dimensional feature vector.

For the HOG, we configure the cell size and block size as (32,4) for the *Butterflies* and *Orchids*; (8,2) for the *Woods* to capture its microscopic structure. Due to the high-dimensional of the obtained HOG features, we apply principal component analysis (PCA) for feature dimensionality reduction. We keep 99% components of the decomposed principal components, which dramatically reduce the obtained feature size.

For the classification model, we use the polynomial kernel SVM. We set the regularisation term as 10 to prevent overfitting of the model.

We also use a shallow CNN architecture i.e., the AlexNet [25] for comparison. In Table 6.4, a CNN architecture without an indication of $*$ denotes that we use a pre-trained network based on a large image datasets which does not include our datasets to extract features. We consequently use the polynomial SVM for classification. The notation $*$ means that we use the strategy presented in this

Table 6.4: Accuracy (%) of different methods on taxonomic datasets

	LBP	HOG	AlexNet	AlexNet*	VGGNet	VGGNet*
Butterflies	91.6±0.3	96.9±0.3	97.8±0.1	99.5±0.1	98.2±0.2	**99.6±0.1**
	82.2±0.2	93.5±0.2	95.1±0.2	98.7±0.1	95.5±0.2	**98.9±0.2**
Orchids	86.0±0.4	88.4±0.5	91.6±0.2	98.4±0.2	92.4±0.2	**98.8±0.2**
	9.4±0.5	41.8±0.7	51.1±1.0	82.7±0.7	50.8±0.4	**86.1±0.5**
Woods	97.2±0.2	75.7 ±0.4	99.2±0.2	100 ±0.0	99.8±0.02	100±0.0
	88.4±0.4	30.1±0.5	85.9±0.4	95.6±0.4	90.7±0.2	**95.6±0.3**

chapter. Namely, we use our datasets to fine-tune the pre-trained network and extend the network with a multi-output layer for prediction.

A2.Results First, from the results, we can see that the taxonomic recognition on a higher rank is relatively more easy than that of a lower rank. This is reflected by a much higher classification accuracy on the level of genus(class) than that on the level of species for all the datasets, using different methods.

Second, if we focus on the well-designed features in the first two columns, we can observe that the LBP can obtain higher recognition accuracy for the *Woods* and the HOG can obtain higher recognition accuracy for the *Butterflies* on both levels. The LBP is advantageous in capturing textural structures and the HOG is capable of holding the whole appearance in an image. Accordingly, the characteristics of the woods are represented as important patterns on the shape and structure of the vessels and tissues; the characteristics of the butterflies are represented in larger scale patterns on butterfly's wings. Those can be separately stressed by the LBP and HOG. For the *Orchids*, the LBP and HOG features obtain similar results on level 1, both of which, however, failed in the species recognition. This is caused by the diverse patterns for the orchids. One should integrate colour, texture, shape as well imaging conditions to characterise orchid's patterns. It is difficult for the LBP and HOG to generalise all these characteristics.

Third, we can find that a simple and pre-trained CNN architecture like AlexNet can obtain better performance on the three datasets than the well-designed features, but it fails to compete with the LBP features on the species recognition of the *Woods*. This is because the pre-trained CNN architectures do not have sufficient training images similar to the *Woods*. This leads the CNN cannot sufficiently generalise the microscopic tissular patterns.

Fourth, after a fine-tuning, both of the shallow and deeper CNN architectures can obtain very accurate recognition on the three datasets. This again illustrates the power of the CNN architecture on representative feature learning for image based taxonomy. In addition, although a small different performance can be found for the two CNN architectures on *Butterflies* and *Woods*, a large improvement is made by the VGGNet for the species classification on the *Orchids*. From the observation, we may conclude that for a relatively simple image base taxonomy which introduces less variant conditions can be solved by a simple CNN architecture such as AlexNet, while the complication of a task, e.g., the taxonomic recognition for the *Orchids*, requires a deeper CNN architecture.

(B) Results visualisation

In this experiment, we further explore the results obtained by the proposed method with the manners of confusion matrix and feature visualisation.

B1. Confusion matrix In Fig. 6.9 (a1) to (a3), we present the confusion matrix for the genus recognition of each dataset obtained from the proposed method, i.e., the VGGNet with a multi-output layer.

In each confusion matrix of Fig 6.9, we use orange lines to indicate the grouping of the species. The species separated by the lines are from the same genus (class). We have left out the species with less than 3 examples and the corresponding result is shown as zero on the diagonal in the confusion matrix. Due to the limited space, we show the names of some selected species.

First, one can observe in the confusion matrices that the recognition accuracy for the three datasets is high which corresponds to the result shown in Table 6.4. Although data imbalance is occurring in the datasets of *Butterflies* and *Orchids*, it is hardly to see serious overfitting for the species with more examples. This can be reflected by the high recall and precision for all the species.

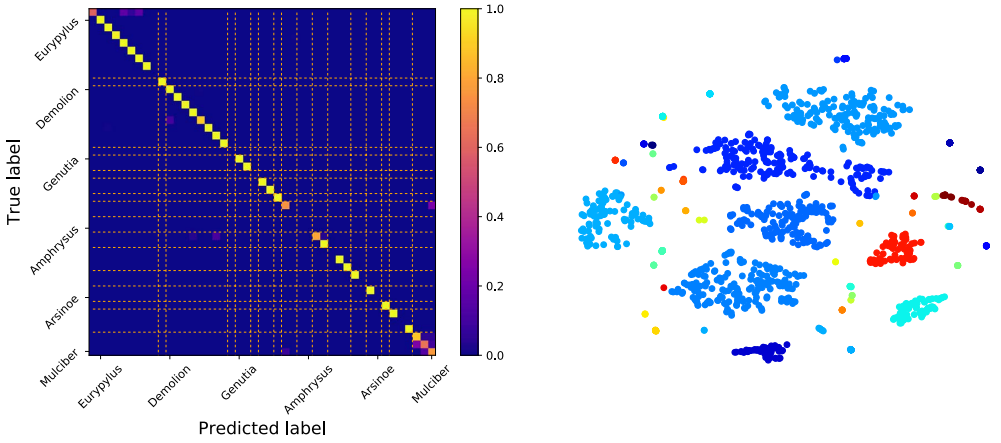
Second, we can find an important phenomenon that the classification errors of the species are mainly distributed within the same genus. One can see the squares associated with the diagonals in Fig. 6.9 (a2) and (a3) for this message. According to this observation, we can conclude that, in the image based taxonomy for biological specimens, it is more difficult to recognise the species which share the same genus. The prediction on a higher level, e.g. genus and class, probably does not help to improve the recognition accuracy on the level of species.

B2. Feature visualisation In Fig. 6.9 (b1) to (b3), we produce the so-called t-SNE map [173] for the visualisation of the representative features obtained in our method. From this visualisation, we can clearly see the separation among different species in each dataset according to the representative features. This is shown as the separated clusters. A relatively sparse t-SNE map is obtained for the *Orchids* and *Woods* due to their large number of species. Another reason resulting in the sparse t-SNE map is that the learned representative features in the same species are very similar for different specimens. This produces rather dense overlap among specimens from the same genus (class). In fact, in each clustering center a dense overlapped with the features extracted from different specimens. Yet, obvious clustering centers can be found for each species in these two datasets.

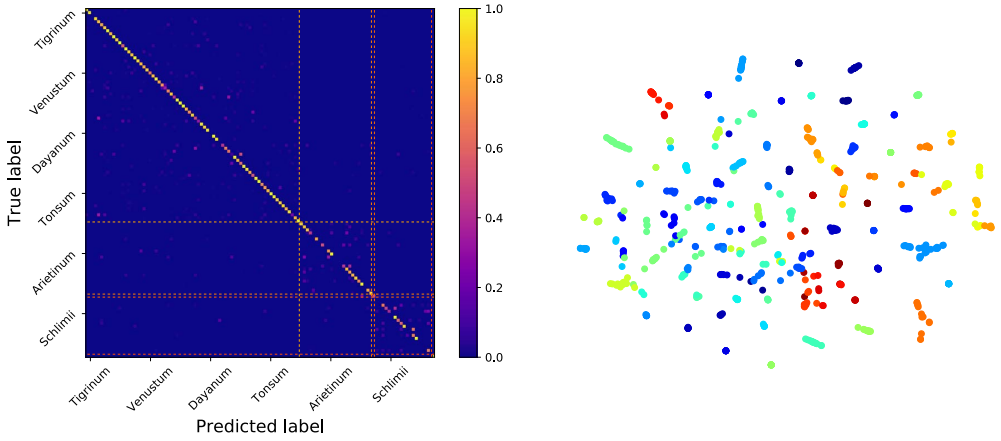
6.2.4 Section conclusions and future work

For the task of image based taxonomy, we have presented a CNN architecture which extends the conventional VGGNet with a multi-output layer. This makes the prediction on each level in the taxonomy possible. We have proposed to apply the fine-tuning strategy to accelerate and stabilise the training of the networks. We also present two taxonomic structured datasets of biological specimens. Compared to the well-designed image features, i.e., LBP and HOG, the proposed method can obtain discriminative and representative features for each task, yielding much better taxonomic recognition accuracy.

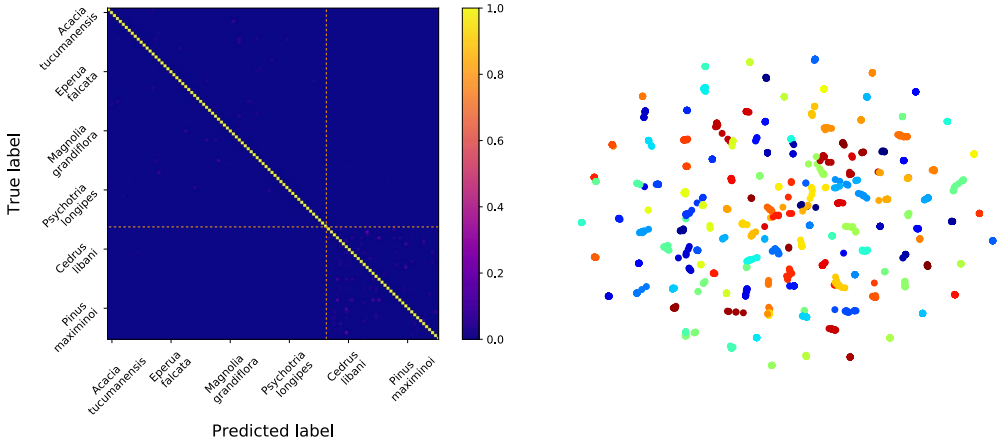
This section answers RQ 6: *To what extent is it possible that the classification models (or regression models) are able to validate the performance of the image features to characterise the phenotypes in support of shape analysis*? It conveys us the message that the CNN architecture is very helpful to characterise the phenotypes including shape and texture from macroscopic to microscopic imaging scale. Importantly, we find that a good estimation on a higher level in a taxonomy probably is not helpful to improve the recognition accuracy on the level of species. In order to further explore this, we can apply a structured prediction model such as the popular CNN+RNN architecture [174]. Regarding the application of phenotype characterisation using microscopy, we need to solve the problem of limited availability of annotated training data. In this context, semi-supervised or weakly supervised learning algorithms should be taken into account. Moreover, an increasing size of the dataset will also help.



(a1) (b1)



(a2) (b2)



(a3) (b3)

Figure 6.9: (a1)-(a3) Confusion matrix and (b1)-(b3) t-SNE map of *Butterflies*, *Orchids* and *Woods* obtained from the proposed method.