Machine Learning Methods for Pollen Image Recognition and Classification

Yubo Feng

August 20, 2015

Abstract

This project investigates and represents several methods that is used for pollen image recognition and classification, including the whole process from feature extraction, selection as well as machine learning based multi-classification methods. The first part of this report will be introduction to the problem and sample sets we used. The second part is explanation of feature extraction, considering geometric features and texture features. The third part will be introduction of methods used to reduce feature attributes and classifiers selection as well as experiments design. The fourth part will be classifier performance and result analysis as well as some interesting observation. The last part will be conclusion.

1 Introduction

1.1 Problem Definition and Importance

Over 20% of all the world's plants are already at the edge of becoming extinct. Saving earth's biodiversity for future generations is an important global task and as many methods as available must be combined to achieve this goal. This involves mapping plants distribution by collecting pollen and identifying them in a laboratory environment. However, pollen grain classification has been an expensive qualitative process, involving observation and discrimination of features by a highly qualified palynologist. It is still the most accurate and effective method. But it certainly limits research progress, taking considerable amounts of time and resources. Automatic recognition of pollen grains can overcome these problems, producing purely objective results faster. Therefore, as we could see, it is necessary to develop good classifier for this problem; recent year, machine learning methods were widely used, different models are built and various approaches were tried, it is still a interesting area.

As described in proposal, this project will focus on recognizing plant category by recognizing pollen image via various machine learning methods. To be more specifically to say, there are a lot of interesting aspects to be explored in this project: (1). What kinds of features we need to extract in order to recognize what type of plant it is? (2). How to extract plant features from pollen images? (3). If all these features are useful? if there exists a set of "minimal feature set" within which the features are enough to recognize this type of plant? (4). What kind of classifier could be used in this question? (5). How about the performances of these classifiers? (6). Where these differences of performance come from? As we may see in below, the frame of this project follows these questions.

1.2 DataSet Describe

In this project, all features should be extracted one by one from raw images, since there is no dataset for pollen features. The raw pollen images will be captured from **Pollen Laboratory**

at the The University of Newcastle, Callaghan, NSW, Australia, which could be found in *http://www.geo.arizona.edu/palynology/nsw/*. Then we extract features by applying methods described in this report.
The image set from Pollen Laboratory contains about 50+ types different geometry characteristics pollen classes, 200 different pollen types, totally about 1000+ pollen images. However, in this project, in order to make things simpler, we only choose 9 types of pollen (# of class), each contains at least 4 images and at most 9 images, average is 6.7 per type; in another word, dataset will contain 60 sample images; the reason why we do in this way lies in that the dataset is no longer a simple txt file, but mixtures of pollen images and titles, which means it takes a lot of time to filter the image and label them; so we only choose some of them to do our project instead of do them all.

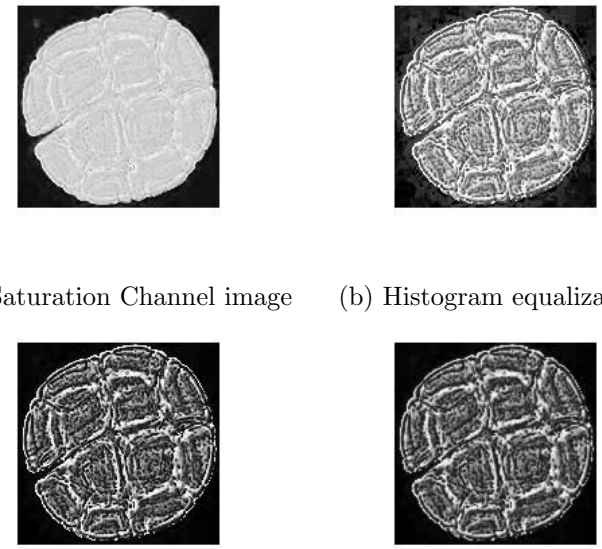
2 Pollen Feature Extraction from Images

In this project, about 46 features are extracted from a single pollen image, geometric features and texture features are considered both. The method used here could be found in several previous work, but we mainly used the methods from [1]. We briefly described the processes as follow:

2.1 Pollen Image Capture

Just as we mentioned, for pollen images, we have to capture them from website one by one, in this step, what we should careful is that we have to make one pollen grain located in the selected region of interest as narrow as possible. The result of this step will be images as **Figure 1**.

Figure 1: Pollen image of Acacia



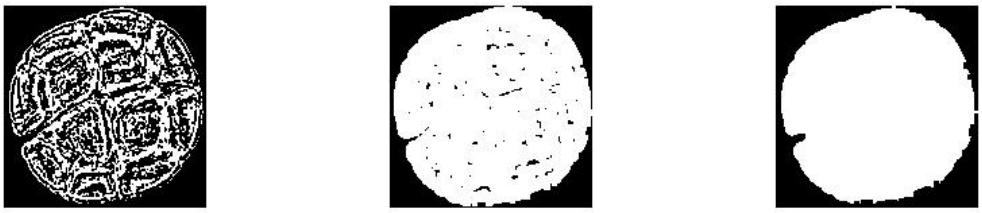
(a) Saturation Channel image (b) Histogram equalization

(c) Gamma filter (d) Gaussian Filter

Figure 2: Pollen precess from Saturation Channel to Gaussian Filter

2.2 Image Preprocessing

- *Stretch*: resize the image to make the image exactly the same for all images, since in following steps, we need to compare them base on the same size of image.
- *Saturation*: isolate the saturation channel of the image, the image value in this channel amount of colour used at each pixel, which could be easy done in HSV image representation.
- *Histogram equalization*: in order to obtain a uniform distribution of the pixel values.
- *Gamma Correction*: in order to make the image correction for threshold. In here we choose $\gamma = 3$, this is a experiment value, see [2].
- *Gaussian Filter*: in order to make the image more insensitive to noise, we filter all noise by Gaussian filter.
- *Binarization*: by setting a threshold, we could get a image consisted of all '1' or '0', then this image contains pure geometric information.
- *Dilation and Hole Filling*: we fill all blank area with white pixels to make it is more easy to obtain object.
- *Erosion*: erase edge areas to make the central object more smoothly
- *Central Connection Item isolation*: sorting by connecting objects areas, the max one is our object, then by applying other objects to 0, we could isolate central item.
- *Mask*: by this step, we filter all back ground but highlight pollen object.



(a) Binarization (b) Holefilling (c) Erosion

Figure 3: Pollen precess from Binarization to Erosion

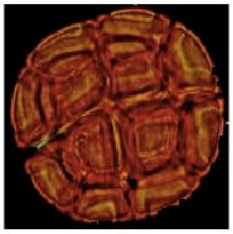


Figure 4: Central Connection Item isolation and Mask

2.3 Image Feature Extraction

In order to get enough image information from the pollen image then to recognize them, we need extract geometric features and texture feature.

2.3.1 Geometric Feature

- *Area*: the amount of pixels with level 1 in the pollen mask.
- *BoundingBox*: Smallest rectangle enclosing the pollen.
- *Centroide*: Refers to the mass centre of the pollen grain.
- *MajorAxisLength*: Length of the major axis of the ellipse with the same second order normalized central moment of the object.
- *MinorAxisLength*: Length of the minor axis of the ellipse with the same second order normalized central moment of the object.
- *ConvexArea*: Area of the smallest convex shape enclosing the object.
- *EquivDiameter*: Diameter of the circle with the same area as the object. ??
- *Solidity*: Portion of the area of the convex region contained in the pollen.
- ?*Perimeter*: Length of the perimeter of the mask image.
- *Extent*: Portion of the area of the bounding box contained in the pollen.

- *Eccentricity*: Relation between the distance of the focus of the ellipse and the length of the principal axis.
- *WeightedCentroid*: This is a centroid computing weighted by the pixel values of the grey-scale image.
- *Shape*: Measures how circular is the pollen. Its values are in the range [0,1], where 1 corresponds to a perfect circle.
- *Box*: These are the coordinates of an inner rectangle area computed from the BoundingBox parameters.
- *Hight*: Length of the largest line enclosed in the pollen.
- *Width*: Length of the largest line enclosed in the pollen and perpendicular to Hight.

2.3.2 Texture Feature

For texture information, we will concentrate on a inner rectangle area computing from the BoundingBox, as we see in **figure 5**. This small area will tell us how pixels are distributed on the image. *The first 4 texture parameters are computed using the grey level co-occurrence matrix (GLCM). This matrix gives information about the frequency of pixel value pairs combinations.*

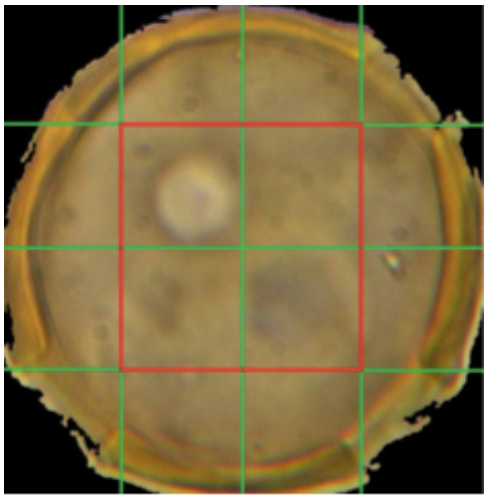


Figure 5: Example of the inner rectangle area computing from the BoudingBox

**the figure are from [1] in order to show concept clearly*

- *Contrast*: Mean intensity difference between a pixel and its neighbours.
- *Correlation*: Measures how must correlated it a pixel with respect to its neighbours.
- *Energy*: Sum of the squared elements of the GLCM.
- *Homogeneity*: Measures how close the distribution of objects of the GLCM are to the diagonal of the GLCM
- *Entropy*: This measure is applied to six different images derived from the original pollen grain image.

- *Fourier Descriptors*: These measures are based on the analysis of the pollen contour points, and it provides information about the pollen shape.
- *Relative areas*: This is a 5 elements vector which values correspond to the number of active pixels (pixels with value 1) after binarizing the pollen image with different thresholds.
- *Relative Objects*: How many objects are there in Relative areas.

After all these process, we could get 46 attributes. One thing should be mentioned is that: some feature contains 2 or more value, for example, *BoundingBox*, we regard this as 2 separate attributes, which make the attributes actually more than features we see. Adding label of this pollen, we totally get 47 attributes to be one simple in our dataset.

3 Feature Reduction, Classifier Selection and Experiment Design

3.1 Feature Reduction

One interesting question in this project mentioned before is that: if all these attributes could be "fundamental" in this problem? in another word, if all these features are suitable for this classification task? There are a lot of methods could answer this question, however, consider complexity and time, we only use 3 kinds of method to reduce our # of attributes.

- we consider Principle Component Analysis (PCA), we try to find enough eigenvectors to cover all attributes in order to make Rebuilt Error as less as possible. Moreover, in order to keep eigenvectors significant, in PCA process, we keep overall variance (or summative variance) is 95% of the original variance. Therefore, 12 eigenvectors could be created, then we can get new dataset base on these eigenvectors, we call it *pollen_pca*.
- we use filtering approach to filter out features with small predictive potential: Correlation coefficients are used, subsets of features that are highly correlated with the class while having low intercorrelation are preferred. After this step, about 16 attributes are selected, we call this attribute set as *pollen_filter*
- wrapper approaches are considered to be another feature selection method: in this approach, Mutual information are used as ranker. The process will be: we rank all attributes, then we will always add the attribute that could decrease our error rate, also, we will use internal cross-validation to do this job. After this step, about 26 attributes are selected, we call this dataset *pollen_wrapper*.
- original data set into our experiment, we call it *pollen*, this set will be regarded as baseline.

After we get these all dataset, we will use these date sets to test classifiers, one thing that is interesting is that, by comparing the performance of various classifiers showed in these dataset, we could see how different features impact on classifiers.

3.2 Classifier Selection

In class, many algorithms are showed that could deal with various type of problems, for this one, considering it is a multi-classification, we cannot use unsupervised learning algorithms, but supervised learning algorithms.
As it is mentioned in proposal, we will use 9 base models to be our classifier: Naive Bayes, Muti-class Logistic Regression with soft function, Neural Network, multi-class SVM, KNN, Best-first Decision

Tree (BFTree), Functional Trees (FT), Logistic Model Trees (LMT), Random Forest (RF).
To make things more interesting, 2 ensemble methods will be tried in this project: both bagging and ada-boosting will be used, and base model will be: NB, Muti-class Logistic Regression, SVM, KNN, RF. The reason for Muti-class Logistic Regression is that this model is very basic; SVM is chosen since it shows good performance in assignments; KNN is non-parametric method, it is used to compare with all other parametric method; RF is the only tree model that is used in ensemble model.
With all these methods, whether it is base model or ensemble model, Naive Bayes always will be baseline, and all models, will be trained and tested on four datasets described above to see the impact of datasets.

3.3 Experiment Design

Experiment will focus on different methods performance on 4 datasets. However, considering the # of samples in dataset are not sufficient to divided into train set and test set (more specifically to say, if we divided into train and test, for some type of plants that have not enough samples may cause overfit), K-fold cross validation will be used to evaluate performance of models. Basic process of experiment is simple: different models will be evaluated on datasets individually.
In this project, machine learning experiment software Weka (*http://www.cs.waikato.ac.nz/ ml/weka/*) will be used to performance our experiments. Weka is a machine learning tool developed by The University of Waikato, New Zealand.

4 Classifier Performance and Result Analysis

For base models, their performance are shown as follow:
From this table, there is some interesting observations:

Table 1: base models performance on 4 dataset, k-fold cross validation

DataSet	Naive Bayes	MLR	NN	SVM	KNN	BFTree	FT	LMT	RF
pollen	62.33	73.00	54.33	78.00	77.33	49.00	76.67	75.67	72.67
pollen_pca	59.00	62.33	53.67	58.00	57.67	43.00	66.00	65.00	63.33
pollen_filter	71.33	74.33	55.00	71.33	78.00	52.00	74.33	73.67	69.00
pollen_wrapper	68.33	79.67	67.00	75.67	76.00	52.00	82.67	79.33	73.67

- BFTree performs worst regardless of impact of dataset, the performance of BFTree even worse than basecase
- FT and KNN shows best performance than other models, means that these two methods is suitable to be used in this problem
- except for pollen_pca, performance of KNN are stable, means that the error rate of KNN are almost keep the same
- all models is failed on pollen_pca dataset, which means if pca method is not good to be used in this problem
- generally to say, models on pollen_wrapper dataset gets better performance than on pollen_bagging

- Naive Bayes performance reaches peak point on pollen_filter among all four datasets, means that if there is too many unrelated attributed, these attributes could become system noise, which will significantly influence NN's performance.

For ensemble models, the performance are show below:

Table 2: pollen

DataSet	Naive Bayes	MLR	SVM	KNN	RF
Bagging	61.67	76.67	76.67	71.67	68.33
Boosting	61.67	73.33	71.67	70.00	76.67

Table 3: pollen_pca

DataSet	Naive Bayes	MLR	SVM	KNN	RF
Bagging	68.33	83.33	55.33	63.33	58.33
Boosting	58.33	58.33	63.33	58.33	61.67

Table 4: pollen_filter

DataSet	Naive Bayes	MLR	SVM	KNN	RF
Bagging	68.33	83.33	68.33	76.67	68.33
Boosting	68.33	80.00	71.67	78.33	73.33

Table 5: pollen_wrapper

DataSet	Naive Bayes	MLR	SVM	KNN	RF
Bagging	65.00	81.67	73.33	68.33	68.33
Boosting	63.33	81.67	73.33	71.67	71.67

From the table, we may find that:

- KNN stay the same performance with base model, it is not hard to think that the performance of KNN is related to nearest neighbors, if we train it by boosting or bagging, since there is no change for one single point's neighbors, then boosting or bagging will not help that much.
- performance of Muti-Logistic regression is significant improved on dataset pollen_filter and pollen_wrapper, all of them $\geq 80\%$, however, in dataset pollen, whether bagging or wrapper cannot improve performance of this model that much; also, in dataset pollen_pca, we could find that boosting even do harm to MLR performance.

5 Conclusion

In this project, pollen image recognition is studied by perform different models on multi versions of abstracted image attributes. There are several lesson to learn:

- not all attributes must be used in classification, some of unrelated attributes may drawback model performance; both filter and wrapper are good methods to be used for feature selection; however, pca reduction method is not a good choice in this problem. If we just use feature reduction, the performance of models cannot worse than full-attribute situation; if we combine ensemble methods with feature selection, better performance is possible: as we see,

especially for regression, if we use feature selection and ensemble methods, a lot of unnecessary computation and system noise will be cut down, the performance is accepted.

- KNN as a kind of non-parametric method, though it is easy and memory cost, but we can see that it is easy to use and the performance is very stable, even for dataset containing redundancy attributes, KNN will give good output sometimes even debate complexity models like NN. From the view of application, KNN is not a bad choice.

- for pollen image classification problem, FT and KNN could used for single base classifier; MLR could be used combined with bagging or boosting for ensemble model.

6 Reference

Mainly Reference

J. Fan and J. Lv, *A selective overview of variable selection in high dimensional feature space*, Statistica Sinica, vol. 20, no. 1, pp. 101 148, 2010.

Marcos del Pozo-Baos, Jaime R. Ticay-Rivas, Jous Cabrera-Falcn, *Image Processing for Pollen Classification*, *Biodiversity Enrichment in a Diverse World*, Chapter 19, ISBN 978-953-51-0718-7, 2012.

Ma da lina Cosmina Popescu, Lucian Mircea Sasu, *Feature extraction, feature selection and machine learning for image classification: A case study*, OPTIM, 2014.

Secondary Reference

M. Langford, G. Taylor, and J. Flenley, *Computerized identification of pollen grains by texture analysis*, Review of Palaeobotany and Palynology, vol. 1, no. 4, pp. 197?203, Nov. 2010.

M. Rodrguez-Damin, E. Cernadas, A. Formella, and P. de S-Otero, *Pollen classification using brightness-based and shape-based descriptors*, ICPR (2), 2004, pp. 212 215.

J. Fan and J. Lv, *A selective overview of variable selection in high dimensional feature space*, Statistica Sinica, vol. 20, no. 1, pp. 101? 148, 2010.

Marcos del Pozo-Baos, Jaime R. Ticay-Rivas, Jous Cabrera-Falcn, *Image Processing for Pollen Classification*, *Biodiversity Enrichment in a Diverse World*, Chapter 19, ISBN 978-953-51-0718-7, 2012.

P. LI, W. J. TRELOAR, J. R. FLENLEY* and L. EMPSON, *Towards automation of palynology 2: the use of texture measures and neural network analysis for automated identification of optical images of pollen grains*, JQS, 2004.