Makeup Or Not

Jiaxin Li [*1] **Meng Zhang** [*1]

1. Abstract

Makeup detection is essential in many scenarios. For example, when a person is wearing makeup, it is difficult to identify and tell who he or she is. In some cases, people are not allowed to wear makeup before entering situations like laboratories. To this end, our project mainly aims to build a classification model to detect makeup from facial images. We conduct various experiments to select features and classifiers in order to achieve higher accuracy. According to the experiment results, HOG and Haar features are most effective to extract information collaboratively from facial images. We also implement the Naive Bayes model as our classifier due to its highest accuracy. The accuracy score on the test set finally reaches 0.5782, which indicates that our model can basically realize the makeup detection function.

2. Introduction

Can you tell if someone is wearing makeup or not? In this project, we propose a makeup detection model that can extract and analyze features from facial images to help you answer the question.

The model can be widely applied - not just for fun, but for practical face recognition uses when wearing makeup is not allowed under some special circumstances. One of the examples is concerning laboratory safety requirements. "As required by the laboratory safety, scientists are not allowed to wear makeup while they are in the lab." (Department of Chemistry and Biochemistry, College of Charleston) "Other examples include no-makeup requirement before certain medical examinations - wearing makeup may violate the diagnosis of instruments and interfere with test results."(Coastal Neurology) Once successfully built, our work is supposed to detect whether people are wearing makeup or not intelligently.

Makeup will also pose heavy interference on face recognition. As indicated by Neslihan in his paper, "the performances of face recognition systems degrade in the presence of makeup on the face."(Kose et al., 2015) To solve this problem, a makeup detection model can work as a powerful pre-processing method, providing makeup labels for subsequent analysis. Based on the makeup detection results, more complicated applications like a makeup removal system can be implemented. In other words, our model will also work as an effective image pre-processing method for face recognition tasks.

In terms of our motivated problem, makeup or not, we have widely reviewed relevant literature and found the existing solutions are not powerful enough. For one, most researches regarding makeup topic utilize only text information to study and analyze cosmetic products. One of the typical examples aims to recommend suitable products according to the shade of foundation. Different from those examples, in our project, we would like to choose an image dataset and to pay more attention to image feature analysis. Another point is that many image recognition studies today are more likely to analyze more common entities, such as cars, animals, paintings and so on. It means our makeup detection work will be an interesting, and relatively new area waiting for more researchers to dive into. Moreover, most of the existing studies only choose one specific model to train the dataset, thus leading to low accuracy. Different from those projects, our project will try to apply different machine learning models and compare the performances between them. The models we consider should include effective ones like decision tree, Naive Bayes, Support Vector Machine, K Nearest Neighbors, Convolutional Neural Network, etc. Then we will choose the best model for our dataset according to the accuracy.

3. Related Work

Makeup Detection "Guo proposed performing correlation mapping between makeup and non-makeup faces on features extracted from local patches. And the authors studied the problem of makeup detection."(Guo et al., 2014) "Chen studied a method to detect the presence of makeup in face images automatically. The proposed algorithm extracted a feature vector that captures the shape, texture and color characteristics of the input face, and employs SVM and Adaboost classifiers to determine the presence or ab-

[*]Equal contribution [1]School of Information, The University of Texas at Austin, Austin, Texas, USA. Correspondence to: Jiaxin Li <lijx@utexas.edu>, Meng Zhang <mz8457@utexas.edu>.

sence of makeup." (Chen et al., 2013) "Kose proposed a facial makeup detector to reduce the impact of makeup in face recognition. The proposed technique extracted a feature vector that captured the shape and texture characteristics of the input face. After feature extraction, two types of classifiers, SVM and Alligator, were applied for comparison purposes." (Kose et al., 2015)

Our work is different from these works because we prepare to use more machine learning models to analyze if a person is wearing makeup or not, including decision tree, Naive Bayes, SVM, KNN, CNN, etc. And we will compare them and choose the best model for our dataset according to the accuracy.

Cosmetic Products Analysis Most researches regarding makeup topic utilize only text information to study and analyze cosmetic products. "Wang constructed a personalized recommender system for the cosmetic business, which incorporates content-based, collaborative filtering, and data mining techniques. Also, the paper introduced a new scoring approach to determine customers' interest scores on products."(Wang et al., 2004) "Pugsee proposed satisfactory conclusion for cosmetic product review comments to analyze the positive and negative information about various cosmetic products by sentiment analysis. The implemented methods used Nave Bay Classifier as a machine learning to automatically classify positive or negative comments on cosmetic products."(Pugsee et al., 2017) "Khraim investigated the influence of brand loyalty on cosmetics buying behavior of female consumers in the Emirate of Abu Dhabi in the UAE. Descriptive analysis, one-way ANOVA and Pearson Correlation were used in the study. The findings indicated that brand name has shown strong correlation with brand loyalty." (Khraim, 2011)

Our work is different from these works because we would like to focus on image feature extraction and analysis rather than text information analysis. Computer vision techniques will be utilized to study features from images.

Objects Detection "Parisi described an experimental system for the recognition of Italian-style car license plates. Characters were classified by a multilayer neural network." (Parisi et al., 1998) "Tang presented a new pattern recognition system to classify large numbers of plankton images detected in real time by the Video Plankton Recorder. Their approach combined traditional invariant moment features and Fourier boundary descriptors with gray-scale morphological granulometries to form a feature vector capturing both shape and texture information of plankton images."(Tang et al., 1998) "Joutou proposed an automatic food image recognition system for recording people's eating habits. In the proposed system, they used the Multiple Kernel Learning (MKL) method to integrate several kinds of image features such as color, texture and SIFT adaptively."(Joutou & Yanai, 2009)

Our work is different from these works because our project focuses on a computer vision topic that is far less studied. Most of the existing studies are more likely to analyze more common entities, such as car plates, animals, paintings, food, etc.

4. Methods

To build a model that can tell if a person is wearing makeup or not, our training process involves data labeling, feature extraction, image classification, and model evaluation. Figure 1 below briefly shows the flowchart of our machine learning system.

We choose a binary classification dataset "Makeup or No Makeup" from Kaggle. This dataset contains 1062 images of people wearing makeup and 444 images of people not wearing makeup. Therefore, the images are labeled as 0 or 1 separately (1 for makeup images, 0 for no makeup images).

In view of the training efficiency, we only use a subset of the whole dataset, which consists of four hundred images randomly selected from each class - 200 for makeup images, 200 for no makeup images.

To make images readable for computer, we utilize feature extraction methods to choose useful information from images and feed these data into our model. After testing different combinations of image features, we choose the combination of two features, namely, the histogram of gradients (HOG) and Haar, as input features. For each image, we concatenate features extracted from the three extraction methods to produce the overall feature.

In order to choose a suitable classifier for our dataset, we test different classification models. Based on their performance, we finally choose Naive Bayes as our classifier, which has higher accuracy among a variety of models.

In terms of model evaluation, we choose the cross validation method to evaluate different models. Accuracy score is calculated as an evaluation metric to compare the performance of different models and image features.

We then examine the performance of our makeup detection model on the rest of the dataset, which includes 862 makeup images and 243 no makeup images. We select 75 percent of images from each class in order to form a training set. The rest of the images are used as our test set. A confusion matrix is utilized to show the classification results. Accuracy score, precision, and recall are calculated as evaluation metrics to examine the performance of the model.

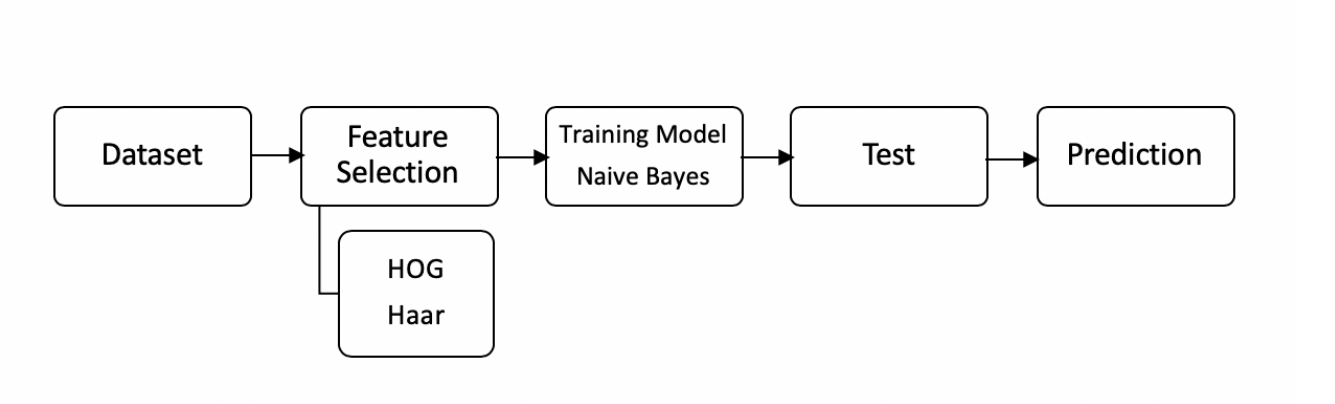

Figure 1. The Flowchart.

5. Experimental Design

5.1. Image Feature Selection

The main purpose of this experiment is to select image features that can be used for feature extraction. In view of the training efficiency, we do not use the whole dataset for this experiment. Instead, we choose only a subset of the whole dataset randomly, which consists of four hundred images - 200 for makeup images, 200 for no makeup images.

There are mainly three effective types of descriptor for makeup detection, including shape, texture and color. When a person is wearing makeup, the three attributes can have great changes. For the shape descriptor, we choose two feature extraction methods, HOG and Canny. "Histogram of gradients (HOG) specifies the magnitude and orientation parameters of feature regions in an image. These parameters are used to determine the objects in the image and image matching." (Öztürk & Bayram, 2018) "Canny is a common method to achieve edge information extraction. Since the application of eye and mouth makeup enhances the local edge structure (corner and contour), an edge orientation histogram is computed." (Chen et al., 2013) In terms of the texture descriptor, we choose local binary pattern (LBP) feature extraction method, which "is used to characterize micro-patterns or micro-structures in the face image by binarizing local neighborhoods based on the differences in pixel intensity between the center pixel and neighborhood pixels, and converting the resulting binary string into a decimal value." (Chen et al., 2013) For the color descriptor, we choose the Haar feature extraction method, which can reflect the image grayscale changes.

We select SVM model as our classifier in this experiment. To test the performance of different feature extraction methods, we firstly use each of them separately as our input to feed the model. Based on the result of each feature selection method, we then concatenate different feature extraction methods together to train our model. The performance of each feature combination will be shown in the result section.

In terms of evaluation, we choose a cross validation method to evaluate our models. Cross validation can be utilized when designing models that generalize well to new, previously unseen examples. Also, cross validation method is good to use all images as training data when we do not have a designated test data. Accuracy score is calculated as the evaluation metric in order to compare the performance of different image features.

5.2. Classifier Selection

This experiment basically aims to choose an appropriate classification model for our makeup detection work. The experimental dataset applied here is the same with that of feature selection experiment - a subset of the whole dataset.

When testing the performance of different models, we firstly extract the same features from images, and then feed them into a variety of models, including Naive Bayes, Decision Tree, K Nearest Neighbor(KNN), Support Vector Machine(SVM), Majority Voting, Bagging and Adaboost. "Naive Bayes is a method to learn through the joint probability of the input features and each class, and then pick the most probable class. Decision tree is a method to learn mapping from input features to class label." (Goodfellow et al., 2016) "KNN is a method to choose K nearest points with the new example through measuring the distance between the new example and training sample, then classify the new example into the class that has the largest probability."(Müller et al., 2016) "SVM is a good method to work with high-dimensional data, which requires little memory and prediction is fast." (Raschka, 2015) The last three methods belongs to ensemble learning method, which can reduce the probability of making a wrong prediction. "Majority

voting is a method that can return the most popular prediction among different classifiers. Bagging is a method to train algorithm repeatedly on different random subsets of the training dataset. Adaboost is a method to value the prediction of each classifier based on the accuracies they have on the training dataset." (Raschka, 2015)

To test the performance of classifiers mentioned above, we firstly hold image feature input stable, that is, the best results get from previous feature selection experiment - the combination of HOG and Haar feature. we then test the performance of different classifiers separately. To conduct majority voting method, we test two different combinations of models separately, that is, the combination o Naive Bayes, Decision Tree and SVM, and the combination of Naive Bayes, KNN and SVM. The combination of classifiers is selected on account of the performance of each classifier separately. We also implement Bagging mechanism on each classifier to figure out if there is an improvement on accuracy score.

The evaluation method applied here is the same as that of feature selection experiment. We keep utilizing cross validation method to evaluate these models. The evaluation metric calculated for each classifier is still accuracy score.

5.3. Dimensionality Reduction

Aiming to overcome the overfitting problem in our project, we also test the Principal Components Analysis(PCA) to reduce feature dimensions. Similarly, we keep using the subset as our dataset in this experiment.

PCA method basically poses data into different dimensions. It is used to drop dimensions that have less explaining ability for variation and preserve the principal dimensions that can account for a large part of variability. In this experiment, we pick four models that stand out in the previous experiment, that is, Naive Bayes, SVM, Bagging(Naive Bayes), and Majority Voting(Naive Bayes, Decision Tree, SVM).

To find the best value of components, we plot the curve of each model, showing the relationship between accuracy score and the number of components. 10 values of components ranging from 1 to 10 are tested in the experiment. We then check the performance of each model with the best components value.

The evaluation method applied here is still the cross validation method to evaluate these models. The evaluation metric calculated for each classifier involves accuracy score.

6. Experimental Results

6.1. Image Feature Selection

Table 1 shows the accuracy performance of different image feature extraction methods using SVM classifier. In view of the cross validation result for each feature, we can tell that HOG feature stands out for its higher accuracy and is more effective when describing our dataset. When concatenating image features together, we find that the performance varies from different combinations. In general, the combination of features will pose a negative effect on accuracy score in most cases. We believe the decrease of accuracy may result from overfitting problem - implementing a complicated feature selection method when we only have limited image samples. Fortunately, we find a slight increase of accuracy score when HOG feature is concatenated with Haar feature - the accuracy score rises from 0.6475 to 0.66.

Table 1. Results for Feature Selection.

Feature	**Accuracy**
HOG	0.6475
Canny	0.5525
LBP	0.5675
Haar	0.5475
HOG+Canny	0.6275
HOG+LBP	0.565
HOG+Haar	0.66
HOG+Canny+Haar	0.6125
HOG+Canny+LBP	0.565
Canny+LBP+Haar	0.565
HOG+LBP+Haar	0.565

6.2. Classifier Selection

Table 2 shows the performance of different classifiers using the combination of the HOG feature and Haar feature. In view of the cross validation result for each classifier, we can tell that Naive Bayes and SVM stands our for their higher accuracy and are more suitable our dataset. Moreover, when utilizing an ensemble learning method in our project, we can tell that Bagging method will slightly improve the performance of Decision Tree and KNN classifier. Majority voting mechanism also improves the performance of Decision Tree, KNN, SVM model. Adaboost does not contribute to an improvement in accuracy score. In general, the Naive Bayes model stands out for its highest accuracy, which is 0.6799. Consequently, we finally implement a Naive Bayes classifier in our project.

6.3. Dimensionality Reduction

Figure 2 briefly shows the PCA curves of the four models tested. From the figure, we can tell that the four models all

Table 2. Results for Model Selection.

Classifier	Accuracy
Naive Bayes	0.6799
Decision Tree	0.505
K Nearest Neighbors	0.515
SVM	0.66
Adaboost	0.5275
Bagging(Naive Bayes)	0.6775
Bagging(Decision Tree)	0.5275
Bagging(KNN)	0.5375
Bagging(SVM)	0.66
Majority Voting(NB+DT+SVM)	0.6699
Majority Voting(NB+KNN+SVM)	0.6575

rich their highest accuracy when the value of components is set as 8. We then check the highest accuracy of all four models and find that the highest accuracy is 0.6475, and is achieved by the Naive Bayes model with 10 principal components. Unfortunately, compared with the results get from our previous experiments, the principal components analysis does not help us a lot to improve the performance of models. As a result of this, we drop the PCA mechanism in our final method.

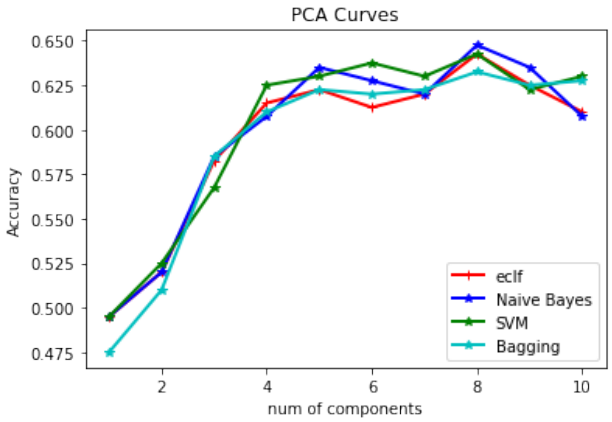

Figure 2. The PCA Curve.

Our work basically explains how to detect makeup from facial images. The performance of our model can still be improved in different ways including dataset cleaning, feature extraction, and model selection. More experiments should be carried out to answer questions involving can our model detect makeup from male facial images and will mechanisms like dimension reduction and regularization help to improve the classification accuracy. Future work for this direction of research should involve extensive studies to improve detection accuracy. "Moreover, higher level facial recognition can be built upon our work to explore methods to remove artifacts introduced by the application of makeup." (Chen et al., 2013)

7. Makeup Detection Performance

The training and testing process in our project have already been introduced in detail in Methods section. As shown by the flowchart, we examine our final model(Naive Bayes classifier, HOG Haar feature) on the rest of the dataset, which includes 862 makeup images and 243 no makeup images. We select 75 percent of images from each class in order to form a training set. The rest of the images are used as our test set.

Figure 3 shows the confusion matrix of the final classification results. Recall that we label makeup image as 1 and no makeup image as 0. From the confusion matrix, we can tell that our model performs better on makeup images than no makeup images. The accuracy of makeup images and no makeup images are 0.586 and 0.55 separately. Accuracy score, precision(marco), and recall(marco) are calculated as evaluation metrics for the whole classification work in order to examine the performance.

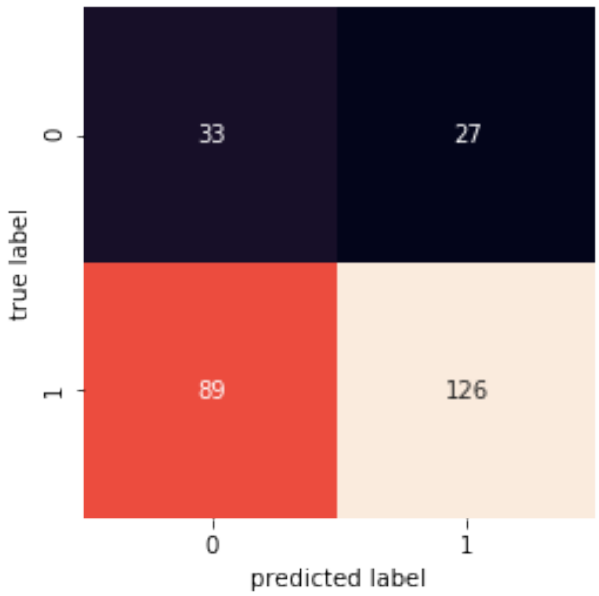

Figure 3. The Confusion Matrix.

Table 3 briefly shows the values of each evaluation metrics. The three metrics all indicate that our model can basically realize the makeup detection function.

Table 3. Model Evaluation.

Evaluation Metrics	Value
Accuracy	0.5782
Precision	0.5470
Recall	0.5680

8. Limitation and Future Work

We have conducted the research and make experiments on the prediction for makeup or not. But we also realize that some limitations lie in our project that can be fixed and improved in the future.

The first limitation refers to the imbalanced dataset applied in our project, which performs in two aspects - the imbalanced size of dataset and the single contents of images. Our dataset consists of two types images, makeup and no makeup, but the number of two classes is not equal. The number of makeup images is almost twice as many as that of non-makeup images. In this situation, sometimes, the accuracy of the model is more likely to be relatively higher. However, the higher accuracy cannot reflect the real case because the machine may make the same prediction for each sample, corresponding to the class that has more data, in order to achieve higher accuracy. In the future, "there are some solutions to combat imbalanced training data, such as collecting more data, evaluating the performance through confusion matrix, resampling dataset, trying different algorithms or penalized models, etc. "(Brownlee, 2015)

Furthermore, our dataset introduces bias into the machine learning system due to low data diversity. Most of images in the dataset are from the Caucasian female. There are only a few images that are from male or female of other races. This situation can make our algorithms less robust because it is difficult for it to make correct predictions for other races. To solve this problem, it is essential to collect more data with higher diversity in the future.

The second limitation is concerned with the quality of dataset in our project. Our dataset is relatively raw, which is not cleaned and preprocessed. For example, some images in our dataset contain two persons, some images even contains a person and a makeup technique. Thus, it is difficult to decide if a person is wearing makeup or not in the image. In the future, when we are faced with a dataset that is relativly small, it is feasible to clean data and crop images manually in order to produce a high-quality dataset. But when the dataset is large, we believe it is important to identify objects in the images in advance, such as mouth, eye, cheek, etc. Then it can be easier and more accurate for machines to decide if the person is wearing makeup or not. Whats more, our dataset exists some classification mistakes. Thus, it is vital to preprocess the data in order to ensure the quality of the dataset.

The third limitation lies in training models. We try different models to train our dataset, including Naive Bayes, Decision Tree, SVM, KNN, Majority Voting, Bagging, and Adaboost. However, we have to admit that the performance of our model is still not good enough and the prediction results are not ideal. To solve this problem, in the future, probably it can be helpful for us to try deep learning methods, for example, using a convolution neural network to learn the image data. Another approach involves utilizing prepossessing methods like Microsoft Vision API. The Vision API will return image features like "color", "text", "face", and other useful tags that are extracted automatically from the images. These features will be helpful for us to conduct makeup detection.

What's more, the goal of our project is makeup detection - deciding whether a person is wearing makeup or not. In the future, there is still much work to improve the project. One idea we find very interesting is doing makeup on facial images automatically. When the machine detects a person that is not wearing makeup, it can provide him or her with the appearance after makeup. The system can even recommend the most suitable appearance for them. Another idea we would like to focus on is removing makeup from facial images automatically. The appearance of people may change a lot after doing makeup, which is quite difficult for others to identify who he or she is. Thus, it is helpful to build a model in order to see the no-makeup appearance.

9. Conclusions

Makeup detection is a computer vision topic that is far less studied. Our work aims to build an effective classification model to automatically detect whether a person is wearing makeup or not. We carried out extensive experiments including feature extraction and model selection on the facial image dataset. Based on the results above, HOG and Haar feature are selected to extract information from facial images. Naive Bayes classifier is implemented when training the model. The established model can be widely applied to real-world makeup detection cases. It can also work as a preprocessing method and successfully improve the performance of face recognition.

References

Brownlee, J. Tactics to combat imbalanced classes in your machine learning dataset. *Machine Learning Mastery*, 19, 2015.

Chen, C., Dantcheva, A., and Ross, A. Automatic facial makeup detection with application in face recognition. In *2013 international conference on biometrics (ICB)*, pp.

1–8. IEEE, 2013.

Coastal Neurology. Preparations for testing. http://coastalneurology.com/cn_prep.htm.

Department of Chemistry and Biochemistry, College of Charleston. Laboratory safety. http://chemistry.cofc.edu/documents/lab-safety-presentation.pdf.

Goodfellow, I., Bengio, Y., and Courville, A. *Deep learning*. MIT press, 2016.

Guo, G., Wen, L., and Yan, S. Face authentication with makeup changes. *IEEE Transactions on Circuits and Systems for Video Technology*, 24(5):814–825, 2014.

Joutou, T. and Yanai, K. A food image recognition system with multiple kernel learning. In *2009 16th IEEE International Conference on Image Processing (ICIP)*, pp. 285–288. IEEE, 2009.

Khraim, H. S. The influence of brand loyalty on cosmetics buying behavior of uae female consumers. *International Journal of Marketing Studies*, 3(2):123, 2011.

Kose, N., Apvrille, L., and Dugelay, J.-L. Facial makeup detection technique based on texture and shape analysis. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, volume 1, pp. 1–7. IEEE, 2015.

Müller, A. C., Guido, S., et al. *Introduction to machine learning with Python: a guide for data scientists*. " O'Reilly Media, Inc.", 2016.

Öztürk, Ş. and Bayram, A. Comparison of hog, mser, sift, fast, lbp and canny features for cell detection in histopathological images. *HELIX*, 8(3):3321–3325, 2018.

Parisi, R., Di Claudio, E., Lucarelli, G., and Orlandi, G. Car plate recognition by neural networks and image processing. In *ISCAS'98. Proceedings of the 1998 IEEE International Symposium on Circuits and Systems (Cat. No. 98CH36187)*, volume 3, pp. 195–198. IEEE, 1998.

Pugsee, P., Sombatsri, P., and Juntiwakul, R. Satisfactory analysis for cosmetic product review comments. In *Proceedings of the 2017 International Conference on Data Mining, Communications and Information Technology*, pp. 13. ACM, 2017.

Raschka, S. *Python machine learning*. Packt Publishing Ltd, 2015.

Tang, X., Stewart, W. K., Huang, H., Gallager, S. M., Davis, C. S., Vincent, L., and Marra, M. Automatic plankton image recognition. *Artificial intelligence review*, 12(1-3): 177–199, 1998.

Wang, Y.-F., Chuang, Y.-L., Hsu, M.-H., and Keh, H.-C. A personalized recommender system for the cosmetic business. *Expert Systems with Applications*, 26(3):427–434, 2004.