# Nose based biometric pipeline

Homework Assignment #3

Image Based Biometrics 2018/19, Faculty of Computer and Information Science, University of Ljubljana

Robert Cvitkovič

***Abstract*—Although there are many different physiological characteristics which are used as modalities for biometric systems there is missing research on the topic of nose biometry. We provide the start for further research by providing a specialized nose testing database and framework. In it we implement some of the common methods used in any biometry pipeline such as detectors, feature extractors and identification models and provide examples of use.**

## I. Introduction

There are many well established physiological characteristics to create an image based biometric system. Examples of those modalities are: ears, iris, faces, ... But all of those have a common drawback. They can be covered by hair, clouding or other fashion accessories. To counter this problem we have to find a modality which is normally not covered and unique enough to be use it in biometry. One such modality is the nose.

There alredy are some studies like [1] and [2] which showed that it's possible to use the nose as an standalone modality but those studies were done in a controlled environment with special equipment. We are more interested in developing a system that works in an uncontrolled environment and uses only images from standard digital cameras. Such work has been done by Zehngut *at. al* [3] who used images from Face Recognition Grand Challenge (FRGC) ver2.0 database[4] and The AR Face database [5] which were taken in and outside of a controlled environment.

In this paper we want to improve the way to test biometric system based on nose image by providing a new dataset of images taken in an uncontrolled environment toughener with manual nose region annotations. We produce a testing framework to unify further researcher in this topic and evaluate two examples of biometric systems build with this framework.

## II. Methodology

Trough the framework we implement the whole pipeline of a typical biometric system. We start by defining a database of images and annotate the nose location. We then continue with nose detection and feature extraction of then nose region. Finlay we use the data to create models to evaluate the accuracy and visualize the results. All this is provided in a python package similar to AWE toolbox [6]. Because of the modularity of the pipeline it's possible to individually change the detector, feature extractor or the mode and thereby study each component individually.

### A. Database

The database consist of images of famous people found on the web. The images contain only one visible nose that is frontal positioned. Sample images can be seen in Figure 1. The database consists of 939 images from 40 different subjects. All images are located in the *data* folder in there corespondent subject sub-folder. All nose locations are denoted in the $info.csv$ file. One line represents one image in the database and contains the image location, x coordinate, y coordinate, width and hight of the nose bounding box all separated by a space.

Figure 1: Samples of images in database.

### B. Annotation tool

The database can be expended with images of current or new subjects. To annotate new pictures we developed a simple graphical user interface (GUI) (Figure 2) which displays new images in the database so the user can select the nose region and save the the selection. A new line in the $info.csv$ is added and the next unannotated image is displayed.


Figure 2: GUI of the annotation tool.

### C. Detection

Detection of the nose region is the most crucial part in the pipeline. If we miss on detecting the nose and select some other part of the image then the rest of the pipeline will have the wrong information and the predictions will be wrong.

In our pipline we implemented the detection based on the Viola-Jones [7] detector. We used the implementation in the open-source OpenCV [8] library which is a widely used image and video processing library. It uses pre-learned cascade models to detect objects in images.

*1) OpenCV cascade:* The library already comes with a grate number of different cascades for detecting a wide wearied of body parts: from full body to face, eyes, ears, etc. It also contains a pre-learned model for nose detection.

*2) Custom cascade:* In addition to the already existing model we decided to create our own. The OpenCV library also includes the tool to do so. We only have to prepare the data. We need two folders: one with pictures of only noses and one with anything else but noses. We therefore created a Python script, which can also be found in the package, that crops the nose and puts it in the nose folder and divides the remaining part of the image and puts the pieces as individual images in the not nose folder. We then run the *opencv_traincascade*[1] to train the model.

### D. Feature extraction

There are many different feature extractors available. We decided to include just two most known: Local binary patterns (LBP) [9] and Histogram of Oriented Gradients (HOG) [10]

### E. Models and Validation

The framework provides two basic models for identifying the subject from the nose. Those two are K-nearest neighbors (KNN) [11] and Support vector machine (SVM) [12].

The model evaluation is done by cross-validation. We also implemented some of the popular scoring measure like accuracy, f1, precision, recall and ROC AUC.

The detector evaluation can also be done with our framework. We use the Intersection over Union (IoU) to compute the detection accuracy score.

$$IoU(A, B) = \frac{A \cap B}{A \cup B}$$

We also implemented two visualization: Receiver operating characteristic (ROC) curve and Cumulative match curve (CMC). One can create multiple pipelines and then visualize all pipelines on the same graph.

## III. Results

We run two separate experiments. We first measured the accuracy of the two previously mentioned detectors. Secondly we compare two pipelines that differ in the feature extractor they use.

### A. Detection

We compare the detections of the OpenCV model and our custom model to the true manually annotated nose detection. The results can be seen in the Table I. We observe better results with the pre-learned model from OpenCV.

[1] https://docs.opencv.org/3.3.0/dc/d88/tutorial_traincascade.html

| model | IoU |
|---|---|
| OpenCV models | 0.596 |
| Custom models | 0.400 |

Table I: Intersection over union for the OpenCV and our custom model

### B. Identification

We compare two pipelines. They both use the true manual annotated nose detections and KNN model but differ in the feature detector. One uses LBP and the other HOG. On Figure 3 we can observe both pipelines CMC. We observe that the HOG based pipeline performs better then the LBP based one.

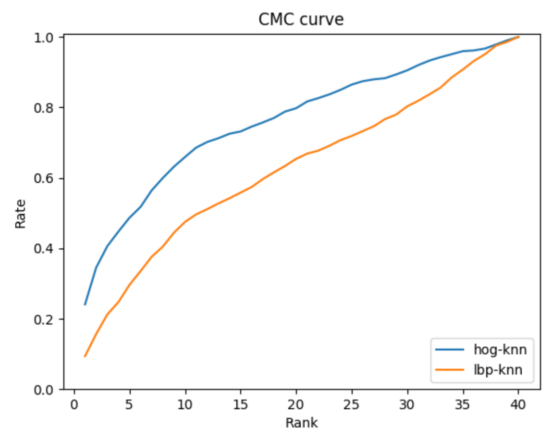


Figure 3: CMC for two pipelines: using LBP feature extractor (orange) and HOG feature extractor (blue)

## IV. Conclusion

In the beginning we observed a clear lack of research on the topic of nose based biometry. We also denote the lack of a standard testing database and and framework. We solved those two problems by providing a new real-world image database and a customizable framework. We also include some examples of using both of them. Now further research in this topic will be easier.

## References


[1] S. Song, K. Ohnuma, Z. Liu, L. Mei, A. Kawada, and T. Monma, “Novel biometrics based on nose pore recognition,” *Optical Engineering*, vol. 48, no. 5, p. 057204, 2009.

[2] H. Drira, B. B. Amor, A. Srivastava, and M. Daoudi, “A riemannian analysis of 3d nose shapes for partial human biometrics,” in *Computer Vision, 2009 IEEE 12th International Conference on.* IEEE, 2009, pp. 2050–2057.

[3] N. Zehngut, F. Juefei-Xu, R. Bardia, D. K. Pal, C. Bhagavatula, and M. Savvides, “Investigating the feasibility of image-based nose biometrics,” in *Image Processing (ICIP), 2015 IEEE International Conference on.* IEEE, 2015, pp. 522–526.

[4] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, “Overview of the face recognition grand challenge,” in *Computer vision and pattern recognition, 2005. CVPR 2005. IEEE computer society conference on*, vol. 1. IEEE, 2005, pp. 947–954.

[5] A. M. Martinez, “The ar face database,” *CVC Technical Report24*, 1998.

[6] Z. Emersic, V. Struc, and P. Peer, “Ear Recognition: More Than a Survey,” *Neurocomputing*, 2017.

[7] P. Viola and M. J. Jones, “Robust real-time face detection,” *International journal of computer vision*, vol. 57, no. 2, pp. 137–154, 2004.

[8] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[9] Z. Guo, L. Zhang, and D. Zhang, "A completed modeling of local binary pattern operator for texture classification," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1657–1663, 2010.

[10] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[11] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992.

[12] J. A. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural processing letters*, vol. 9, no. 3, pp. 293–300, 1999.