Assignment #5: (Biometric Pipeline)

Matjaž Zupančič Muc
IBB 22/23, FRI, UL
mm1706@student.uni-lj.si

I. Introduction

In this work, we implement a ear recognition pipeline. Our pipeline includes training a custom DeepLabV3 segmentation model to isolate ears in images, extracting both Local Binary Patterns (LBPs) and ResNet50 (pre-trained on ImageNet) features from the segmented ears, and implementing a matching stage to identify individuals based on extracted features. The performance of our system is evaluated at each stage of the pipeline, including the segmentation, recognition, and overall pipeline stage.

II. Related Work

[1] presents a method for automated glomerulus detection in transmission electron microscopy (TEM) images using a combination of convolutional neural networks (CNNs) and Local Binary Patterns (LBPs). The method involves the fusion of CNN and LBP maps at multiple scales to capture both the global and local features of glomeruli in TEM images. The performance of the proposed method was evaluated on a dataset of TEM images and was found to outperform other approaches for glomerulus detection, including the use of CNNs alone and the use of unsupervised learning techniques. [2] presents a method for face recognition using a residual convolution neural network (ResNet). The proposed method involves training a ResNet on a dataset of face images to learn features for face recognition. The performance of the proposed method was evaluated on several public face recognition datasets and was found to outperform other state-of-the-art methods. These findings demonstrate the effectiveness of using a ResNet for face recognition.

III. Methodology

Implemented pipeline consists of the following stages: segmentation stage, normalization stage, feature extraction stage and matching stage. Figure 1 shows the implemented pipeline in great detail, we can see that the pipeline is split into two parts: the enrollment pipeline and the recognition pipeline. The segmentation stage implements a DeepLabV3 segmentation model ([3]). DeepLabV3 is a fully convolutional network (FCN) which uses the so called Atrous Convolutions and Atrous Spatial Pyramid Pooling (ASPP) modules. Atrous convolution is an alternative for the down-sampling layer, it increases the receptive field whilst maintains the spatial dimension of feature maps. The fact that the spatial dimension is preserved helps us with the task of semantic segmentation which requires detailed spatial information. We used the PyTorch implementation of DeepLabV3 (https://pytorch.org/hub/pytorch_vision_deeplabv3_resnet101/) which is trained to classify pixels into 21 classes. We start by swapping the head of the model to classify pixels to a single class (i.e the ear class). Since the output of the model is a set of logits, i.e each pixel value is in range $(-\infty, \infty)$, we add a Sigmoid function $\sigma(x) = \frac{1}{1+e^{(-x)}}$ which squashes the model outputs in range $[0, 1]$. Each pixel value now represent a probability of it being an ear. We train the model using binary cross entropy (BCE). The model is trained using the following dataset (https://tinyurl.com/ibbdataset). Dataset consists of 14276 images. On each image there is a single person with a visible left or right ear. The person is in a arbitrary position and is arbitrary away from the camera. We split the dataset into train (80% of data) and test set (20% of data). The test set is further split into a validation set (20% of train data). Before training the model images are re-scaled to the size of (300×300) and normalized to the range expected by the DeepLabV3 model. The model is trained for 10 epochs using the Adam optimizer. Figure 2 shows the loss as a function of epochs, we can see that the model learns fast since the loss after the first epoch is around 0.01. In the normalization stage the segmented ear is cropped from the image and normalized to values in range expected by the ResNet50 model (used in the feature extraction stage). The image is cropped to a shape of (128×128). The feature extraction stage extracts two sets of feature vectors. The LBP feature vectors and LBP + ResNet50 feature vectors. We extract LBP features, using number of points $P = 8$ and radius $R = 1$. Using the LBP feature map we compute histograms in each tile of shape (16×16), histograms are than concatenated and normalized. Each LBP feature vector is therefore of shape $(1 \times 2^8 * \#\text{tiles})$. LBP + ResNet50 feature vectors are constructed by concatenating LBP feature vectors and ResNet50 features. ResNet50 is a CNN trained for image classification (proposed in [4]). ResNet50 has been trained on the ImageNet dataset, which consists of over 1 million images and 1000 classes. To obtain ResNet50 features we pass the cropped and normalized ear image through the model and obtain a feature vector of shape (1×2024). The concatenated feature vector is therefore of shape $(1 \times 2^8 * \#\text{tiles} + 2024)$. Feature vectors are extracted and saved in the enrolled database for all samples in the train dataset. In the matching stage we compute the distance between current feature vector and all feature vectors in the enrolled in the database. The predicted identify is the identity

from the database which has the smallest distance to the current feature vector.

IV. Experiments

We used the the following dataset (https://tinyurl.com/ibbdataset). As mentioned the dataset consists of 14276 images. On each image there is a single person with a visible left or right ear. The person is in a arbitrary position and is arbitrary away from the camera. We split the dataset into train (80% of data) and test set (20% of data). All samples from the train set are used to train and validate the segmentation model. We extract features from samples in the train set and save them to the database. Test samples are used to evaluate the segmentation model, the recognition stage and the whole pipeline. The segmentation stage is evaluated using the mean Intersection Over Union (IOU), we also compute a histogram of IOUs. Since the model returns a probability of each pixel being an ear, we define a threshold $\tau = 0.5$ above which we classify a pixel as an ear pixel. We take a look at the best and the worst segmentation's. The recognition stage is evaluated using the rank 1 and rank 5 accuracy. To evaluate the recognition stage we bypass the segmentation stage and use ground truth segmentation masks as the input to the normalization stage. The whole pipeline is evaluated using rank 1 and rank 5 accuracy. To evaluate the whole pipeline we use the segmentation model to segment ears for all samples in the train and test set. We inspect the histogram of mean probabilities returned by the model and use the segmentation threshold of 0.25 in the segmentation stage of the pipeline. The feature extraction stage uses LBP with the following parameters: number of points $P = 8$, radius $R = 1$ and tiles of shape (16×16). As a baseline we use a random model, which selects a set of random individuals from the enrolled database. We compute the rank 1 and rank 5 accuracy of the random model.

V. Results and Discussion

A. Results

B. Discussion

Figure 4 shows a histogram of IOUs over test samples, to binarize the predicted segmentation mask we use a threshold $\tau = 0.5$ (i.e each pixel in the model output above τ is considered an ear and denoted with 1). We can see that in general the model performs well on the test set, since most of the predictions have an IOU above 0.8. Figure 5 shows a histogram of average non-zero probabilities, predicted by the model. We can see that most predictions have a average probability above 0.25. Figure 3 shows a set of three predictions with the highest IOU and a set of three predictions with the lowest IOU. The segmentation's masks were binarized using the threshold $\tau = 0.5$. We can see that the model performs great when ear is clearly visible and is of high enough resolution. On the other hand the model struggles when the person is far away from the camera and / or the ear is not clearly visible (last picture). Table I shows the performance of the recognition stage. We can see that the best rank 1 and rank 5 is achieved using the cityblock distance metric on top of LBP feature vectors. We also see that LBP features outperform the LBP + ResNet50 features. Same can be observed on Table II which shows the performance of the whole pipeline. In general it seems like ResNet50 pre-trained on ImageNet is not able to accurately encode the (per

TABLE I

Table shows the rank 1 and rank 5 accuracy for the recognition part of the pipeline. We vary the mode (i.e the type of feature vector) and distance metric used in the matching stage.

Mode	Distance metric	Rank 1 [%]	Rank 5 [%]
LBP	euclidean	15.35	26.21
LBP	cosine	17.31	29.09
LBP	cityblock	23.04	36.46
LBP + RESNET50	euclidean	16.10	27.13
LBP + RESNET50	cosine	16.66	27.95
LBP + RESNET50	cityblock	20.62	34.46

TABLE II

Table shows the rank 1 and rank 5 accuracy for the whole pipeline. We vary the mode (i.e the type of feature vector) and distance metric used in the matching stage.

Mode	Distance metric	Rank 1 [%]	Rank 5 [%]
LBP	euclidean	12.96	21.37
LBP	cosine	14.63	23.40
LBP	cityblock	19.41	29.88
LBP + RESNET50	euclidean	12.76	23.20
LBP + RESNET50	cosine	13.48	23.01
LBP + RESNET50	cityblock	16.53	29.49

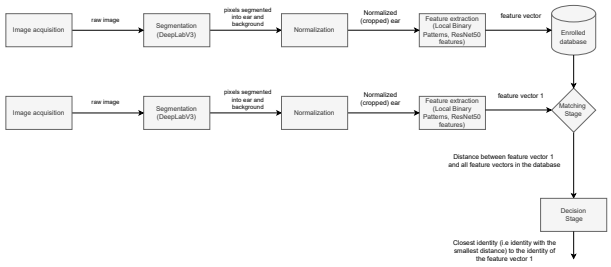

Fig. 1. Ear recognition pipeline. The pipeline is split into enrollment and recognition pipeline, both consist of the following stages: segmentation stage, normalization stage, feature extraction stage and matching stage.

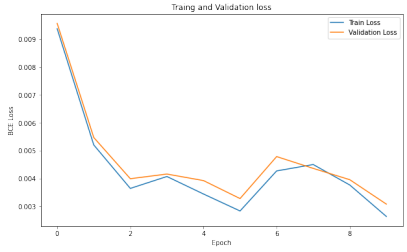

Fig. 2. Binary Cross Entropy (BCE) loss as a function of epochs. The blue line represents the loss computed on the training set, the orange line represents the loss computed on the validation set.

TABLE III
TABLE SHOWS THE RANK 1 AND RANK 5 ACCURACY FOR A RANDOM MODEL.

Mode	Rank 1 [%]	Rank 5 [%]
Random	0.19	1.14

identity) ear information. We also see the performance of the recognition stage is greater by a factor of around $1,19$. Finally Table III shows random model performance.

VI. CONCLUSION

The results of the experiment show that the segmentation model performs well on the test set, with most predictions having an IOU above 0.8. The model performs best when the ear is clearly visible and of high resolution, but struggles when the person is far from the camera or the ear is not clearly visible. In the recognition stage, the best performance of $rank1 = 23.04$ and $rank5 = 36.46$ was achieved using the cityblock distance metric on LBP feature vectors, and LBP features outperformed LBP + ResNet50 features. The whole pipeline also performed better with LBP features and the cityblock distance metric (best performance $rank1 = 19.41$ and $rank5 = 29.88$). It appears that the ResNet50 pre-trained on ImageNet is not effective at encoding (per identity) ear information. The system has a large number of hyper-parameters which could all be further explored and evaluated, for example we could vary the hyper-parameters of the LBP feature extractor, we could vary the threshold τ of the segmentation stage. We could also fine-tune the ResNet50 to classify between individuals, i.e learn the feature extraction and the classification task.

REFERENCES

[1] E. Wetzer, J. Lindblad, I.-M. Sintorn, K. Hultenby, and N. Sladoje, "Towards automated multiscale imaging and analysis in tem: Glomerulus detection by fusion of cnn and lbp maps," in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018, pp. 0–0.

[2] A. Husain and V. P. Vishvakarma, "Face recognition method based on residual convolution neural network," in *IOP Conference Series: Materials Science and Engineering*, vol. 1228, no. 1. IOP Publishing, 2022, p. 012005.

[3] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation. arxiv 2017," *arXiv preprint arXiv:1706.05587*, vol. 2, 2019.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

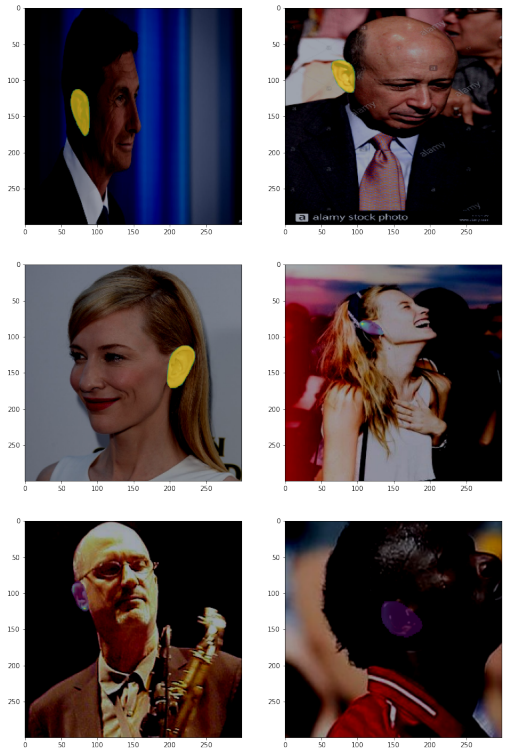

Fig. 3. Figure shows a set of three good and poor segmentation's. The first three segmentation's have an IOU of around 98.5%, the last three have an IOU of around 0.48%. Note that to binarize predictions and compute the IOU we used a threshold $\tau = 0.5$.

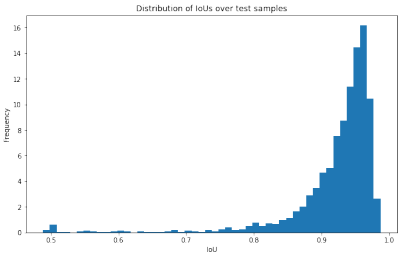

Fig. 4. Figure shows a histogram of IOUs over the test dataset. To binarize predicted probabilities to either 0 or 1 we use a threshold of $\tau = 0.5$.

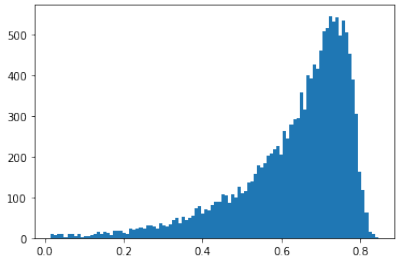

Fig. 5. Figure shows a histogram of mean probabilities returned by the segmentation model. Non-zero predicted probabilities are averaged for each image in the train data-set, using the average probabilities a histogram is computed.