Unconstrained Ear Recognition Using Residual Learning and Attention Mechanisms

Tim Oblak, Blaž Meden, Peter Peer, Žiga Emeršič

Faculty of Computer and Information Science, University of Ljubljana, Slovenia
E-mail: to1702@student.uni-lj.si, {blaz.meden, peter.peer, ziga.emersic}@fri.uni-lj.si

Abstract

With the recent popularity of deep convolutional neural networks, image-based biometrics is one of many domains, that consequently gained new progress in solving previously incomplete or unanswered challenges. While some biometric modalities, like the fingerprint, are already considered mature, others are still in need of more reliable approaches. Ears can be used for person identification since they have the necessary properties of a biometric modality. High accuracy ear identification systems do exist but mostly focus on a controlled environment. In this paper, we try to improve the current state-of-the-art in ear recognition by using residual learning and attention mechanisms. By stacking residual building blocks, we find the optimal architecture to be ResNet with 18 convolutional layers. We achieve a Rank-1 score of 54.46% with full model learning, which is a 5.35 percentage point improvement from the previous best trained on the VGG architecture, however, the model still underperforms against those, trained with selective learning. We observe that aggressive data augmentation is needed when dealing with a small dataset. We also conclude that the Attention Model performance is subpar compared to other architectures.

1 Introduction

In recent years, the use of deep neural networks in the broader field of machine learning brought us new achievements in a diverse range of challenges. Especially, the field of computer vision gained significant improvements with image recognition and object detection using deep *convolutional neural networks (CNN)*. They are known to have good generalization properties in an unconstrained environment. This includes images captured in the wild or on the go, which often contain noise, occlusion, different object rotations and lighting conditions. Such a property is especially useful in forensic, security or surveillance scenarios, where environment impact on the captured image is nondeterministic.

The application of deep learning is also progressively being used in the domain of biometrics, specifically with identification using certain biometric modalities, where until recently, expert knowledge or conventional computer vision local descriptors were used to tackle the problem. A biometric modality can be represented as a biological trait of a living being, which is unique and can be attributed to the being's identity. While recognition of some modalities, such as the fingerprint, are already considered mature, there are still those, that need better approaches to be reliable in practice.

In this paper, our main focus is ear recognition. As stated in [8], ears are known to have rigid biometric properties, such as uniqueness, longevity and a persistent shape, which does not change drastically throughout our lifetimes. Even though ear recognition systems have achieved some maturity, their success is still dependant on a controlled indoor environment.

In comparison to modalities such as eyes, face or fingerprints, there is still a lack of large-scale datasets available to the public. Even existing ones are usually small or only contain images captured in a constrained environment. With this in mind, we make use of data augmentation techniques to artificially inflate our training data.

To summarize, our first and foremost goal is to implement and test two specific deep neural network architectures that proved to achieve state-of-the-art results on some of the biggest public datasets, such as ImageNet [5], and compare them to the current best-performing architectures in the domain of ear recognition. In the process, we also want to measure the impact of data augmentation aggressiveness on the model performance.

2 Related work

A few extensive surveys have been made in this domain. In *A Survey on Ear Biometrics* [8], the authors make an extensive overview of detection techniques, feature extractors and available datasets, while providing experimental results and listing other open research questions. The report *Rank-1* identification rate ranges from 72.7% to 100%. Note, that all mentioned datasets are acquired in a controlled or semi-controlled environment. No deep neural network approaches were used. In *Ear Recognition: More Than a Survey* [18], the authors mostly focus on the recognition aspect. Their summary of known approaches lists *Rank-1* scores similar to the previously mentioned survey [8]. They also propose the *Annotated Web Ears (AWE)* dataset, which contains 1000 images of 100 persons captured on the internet and thoroughly annotated. It is an unconstrained dataset which represents real-world conditions. Again, none of the mentioned ap-

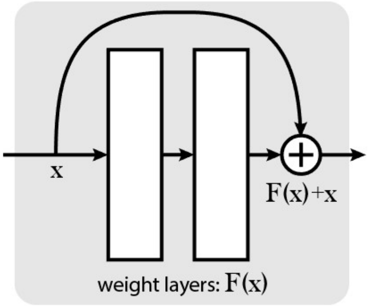

Figure 1: A residual building block used in ResNet and Residual attention network architectures.

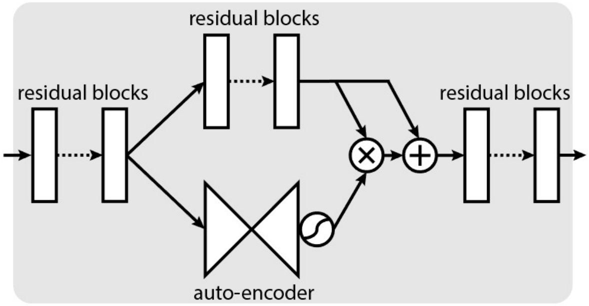

Figure 2: The attention module building block of the Residual Attention Network.

proaches outlined in the paper use any sort of deep learning. Roughly, the usual ear recognition approaches can be divided into 4 groups:

Geometric approaches focus on the extraction and analysis of the geometrical features, such as outer curves and helix point location. They rely on edge detection as their pre-processing step, so all of the textual information is therefore lost and not used for recognition. They are, however, usually scale and rotation invariant.

Holistic approaches encode the input image ear representations as a whole. They are prone to error when varying illumination or different pose of the subject is presented in the input image.

Local approaches make use of local neighborhood descriptors. Opposed to geometric approaches, they do not rely on relations between different points on the image, but between neighboring pixels in a small area. The descriptors can be calculated densely across the image or be selected beforehand for their uniqueness.

Deep-learning-based approaches mostly use CNNs to build the recognition model. They are usually learned end-to-end, which simplifies the recognition pipeline, however, the number of trainable parameters is higher in comparison to other machine learning algorithms. In [16], the authors achieved state-of-the-art results using deep learning with CNNs. They used full model learning and selective learning on top of features learned from the ImageNet dataset. A *Rank-1* score of 62.0% and a *Rank-5* score of 80.46% were achieved on the AWE dataset.

Lastly, different hybrid approaches also exist. These represent any combination of all above-listed approaches.

3 Methodology

In this section, we make an overview of the techniques we use to design and test new architectures. First, we present the main ideas behind existing architectures and then describe our workflow.

3.1 Deep Residual Networks

The residual learning approach was first published by Microsoft Research Asia [11, 12]. The main idea behind this architecture lies in its core building block, called the residual block. The block is shown in figure 1. It has two convolutional layers, where each of them also includes a batch normalization layer and an activation layer. The input and output of the block are connected with a ”shortcut connection” which performs an identity mapping and its output is added to the output of the stacked layers. With this mechanism, the blocks that contribute to the increasing loss can be skipped. This allows the network to be even deeper without suffering from accuracy oversaturation and rapid degradation. These ”shortcut connections” do not add any new parameters or computational complexity and can be used alongside common optimization techniques without changing the backpropagation calculations. Similar ideas were found previously in LSTM [10] and Highway Networks [9] and mostly used in a *recurrent neural network (RNN)* settings. The authors have published multiple versions, which include 50, 101, 152, 200 and even 1001 layer architectures.

3.2 Residual Attention Network

The Residual Attention Network is a CNN which is mostly made from previously mentioned residual block but also includes a truncated branch, which implements the attention mechanism. This mechanism was until recently mostly popular with RNN architectures in natural language processing and machine translation domains, where new samples are weighted based on the occurrence in previous iterations. The authors propose the Attention Module structure, shown in figure 2, which generates attention-aware features. The network is then constructed by stacking multiple Attention Modules.

Each of these modules can be divided into two branches; mask branch and trunk branch. The function of the trunk branch is to perform the usual feature processing and can be adopted into any network structure. The authors chose to use the residual block as their structure of choice. The mask branch is structured in form of an auto-encoder, also found in other architectures [3, 4]. The output of the mask branch is a soft mask, which, when joined with trunk branch output, acts as sort of a control gate function for every neuron output of the trunk branch.

The main motivation for using attention mechanisms lies in their ability to focus on a subset of inputs. This could be helpful in case of occlusion by hair or ear accessories.

3.3 Experiments

We devise a plan to test the described architectures in full. First, we test the effect of data augmentation. Best performing augmentation rate will be used as the baseline. Then, we implement and test the residual attention network. There, some of the residual blocks are substituted with the Attention Modules in a way, that the final depth remains unchanged.

As of now, there is no pre-trained ImageNet model available for any of these architectures. Because of the lack of resources we are therefore not able to do selective learning and are only comparing the full model scores.

4 Experimental Setup

Here, we define our work environment including tools, datasets, performance metrics and the choice of hyper-parameters.

4.1 Dataset selection & augmentation

For our dataset, we use the images provided by the *Unconstrained Ear Recognition Challenge (UERC)* [17]. We divide the dataset into groups of 40% and 60% of images for the test and train data, respectively.

Since the size of the dataset is relatively small, generic data augmentation [7] is used to artificially inflate our training set. We make multiple augmentation configurations, which include no augmentation, $10\times$ and $100\times$ the augmentation rate for each image in the training set.

Images are first resized to a shape of $224 \times 224 \times 3$, which is the shape of the network input. Then, we follow the randomized augmentation process described in [16]. Before entering the network, images are normalized to a range between -1 and 1 and individual channel mean values are subtracted.

4.2 Performance metrics

To evaluate and compare trained models, we adopt the standard recognition performance metrics. *Rank-1* and *Rank-5* recognition rates are computed. For *Rank-1*, we check if the prediction with the highest probability matches the ground truth label. For *Rank-5*, we check if predictions with top five probabilities contain the ground truth label.

We evaluate the scores graphically by plotting the *Cumulative Match Curve (CMC)*. The graph shows the recognition rate of all possible ranks. We also compute the *Area Under Cumulative Match Curve (AUCMC)*, which summarizes the visual CMC numerically.

4.3 Research environment

Our experiments are made in *Python 3.4* programming language, and we use *Keras 2.0.5* [2] with a *TensorFlow 1.4.0* [14] backend as our main neural network library. We train and test the setup on a *ubuntu 16.04* computer, with a *GeForce® GTX 1080 Ti* graphics card, *Intel® Core*^{TM} *i7-7700K* processor and *16 GB* physical memory.

4.4 Hyper-parameters

For the optimization algorithm, we use Adam [1]. Compared to stochastic optimizers such as SGD, it converges faster and has more momentum configuration options. It also requires less fine-tuning and can be set by only specifying the learning rate. We use a learning rate of 0.001, but reduce it by half when the loss function starts to converge. We use categorical cross entropy for computing the loss of the last softmax layer.

Table 1: Effect of different data augmentation rates applied to dataset. Models are trained using the ResNet-18 architecture.

	Rank-1 [%]	Rank-5 [%]	AUCMC [%]
No aug.	31.32	51.77	89.06
10x	36.70	56.62	91.18
100x	**54.46**	**73.19**	**95.24**

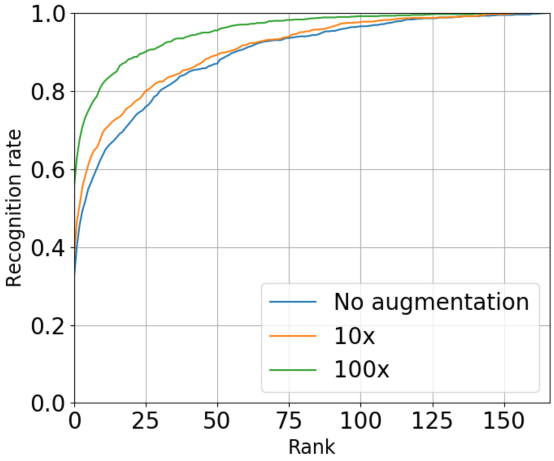

Figure 3: Cumulative Match Curves of ResNet-18 models with different data augmentation rates applied.

A custom data generator is used to supply images to the network in batches of 64. We do not set a fixed epoch number, but implement an early stopping feature, which stops the learning phase when loss function converges for some time.

5 Results & Discussion

In this section, we present and discuss our results regarding architecture depth, type and the amount of data augmentation present.

5.1 Residual Network & data augmentation

The preliminary results have shown that, when comparing network depth, the 18-layer ResNet architecture performed best on the *UERC* dataset. We, therefore, used this configuration as our baseline. Also, data augmentation has a significant impact on the final model performance, as it is presented in table 1. The $100\times$ augmentation rate outperforms non-augmented dataset by 23.14 Rank-1 percentage points achieving a score of 54.46%. Cumulative match curves are shown in figure 3. With data augmentation, we fill the intra-class variance gaps caused by images gathered in the wild. This leads to better generalization, even in different environmental conditions.

5.2 Residual Attention Network

For our last experiment, we implemented the Attention Module from [13]. The architecture was then designed empirically by modifying the baseline ResNet-18. Here, we describe the best performing model. Again, we test the architecture with multiple data augmentation rates, but the configuration with $100\times$ data augmentation rate produced best results.

As presented in table 2, the model performed subpar compared to ResNet-18. It achieved a Rank-1 score of 37.67%. The best *Rank-1* score would only be comparable to ResNet-18 model with no data augmentation applied.

Table 2: Performance comparison of different ear recognition CNN architectures, trained with full model learning. We reference VGG as the previous-best model.

	Rank-1 [%]	Rank-5 [%]	AUCMC [%]
VGG	49.08	66.67	92.99
ResNet	**54.46**	**73.19**	**95.24**
RAN	37.67	60.27	92.66

5.3 Final comparison

To summarize our findings, the results in table 2 illustrate, that ResNet-18 architecture outperforms other models, trained with full model learning. In [16], best results were achieved by using SqueezeNet [6] architecture, which achieved a *Rank-1* score of 62.00%, but was trained using selective learning, thus not comparable to our methods. We do, however, include the results achieved by VGG [15] architecture, as it performed best when trained by full model learning.

6 Conclusion

There are still many domain-specific challenges, that could be improved with the use of deep learning and ear recognition is definitely one of them, as we demonstrate with our experiments. The performance of standard computer vision and expert knowledge approaches is well documented in multiple extensive surveys, while there are still various possible improvements to be made with the use of CNNs.

We have applied two novel CNN structures to the area of ear recognition, ResNet and its modification using the Attention Module. We first illustrated the importance of aggressive data augmentation rate. Significant improvements in performance were measured when such an augmentation was applied.

We improved the previous-best unconstrained ear recognition model, which was also trained using full model learning. Note, that training models with selective learning using pre-trained weights still outperforms our methods by a margin of $\tilde{8}$ percentage points. On a dataset with 166 subjects and 2,304 cropped ear images, our best model achieved *Rank-1* and *Rank-5* scores of 54.46% and 73.19%, respectively.

In the future, we will train the ResNet-18 architecture using selective learning, as we believe that state-of-the-art unconstrained ear recognition results could be achieved.

References

[1] D. P. Kingma, J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[2] F. Chollet. Keras, 2015. Available at: https://github.com/fchollet/keras.

[3] V. Badrinarayanan, A. Kendall, R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *CoRR*, abs/1511.00561, 2015.

[4] A. Newell, K. Yang, J. Deng. Stacked hourglass networks for human pose estimation. *European Conference on Computer Vision*, pages 483–499, 2016.

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei. ImageNet: A large-scale hierarchical image database. *Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[6] F. N. Iandola, M. W. Moskewicz, K. Ashraf, S. Han, W. J. Dally, K. Keutzer. Squeezenet: Alexnet-level accuracy with 50× fewer parameters and 1 MiB model size. *CoRR*, abs/1602.07360, 2016.

[7] L. Taylor, G. Nitschke. Improving deep learning using generic data augmentation. *CoRR*, abs/1708.06020, 2017.

[8] A. Abaza, A. Ross, C. Hebert, M. A. F. Harrison, M. S. Nixon. A survey on ear biometrics. *ACM Computing Surveys*, 45(2):1–35, 2013.

[9] R. Srivastava, K. Greff, J. Schmidhuber. Training very deep networks. *Advances in Neural Information Processing Systems*, pages 2377—2385, 2015.

[10] S. Hochreiter, J. Schmidhuber. Long short-term memory. *Neural Computation*, pages 1735—-1780, 1997.

[11] K. He, X. Zhang, S. Ren, J. Sun. Deep residual learning for image recognition. *Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[12] K. He, X. Zhang, S. Ren, J. Sun. Identity mappings in deep residual networks. *European Conference on Computer Vision*, pages 630–645, 2016.

[13] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, X. Tang. Residual attention network for image classification. *Conference on Computer Vision and Pattern Recognition*, pages 6450–6458, 2017.

[14] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Available at: https://www.tensorflow.org/.

[15] K. Simonyan, A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[16] Ž. Emeršič, D. Štepec, V. Štruc, P. Peer. Training convolutional neural networks with limited training data for ear recognition in the wild. *International Conference on Automatic Face Gesture Recognition*, pages 987–994, 2017.

[17] Ž. EmerŠic, D. Štepec, V. Štruc, P. Peer, A. George, A. Ahmad, E. Omar, T. E. Boult, R. Safdari, Y. Zhou, S. Zafeiriou, D. Yaman, F. I. Eyiokur, H. K. Ekenel. The unconstrained ear recognition challenge. *CoRR*, abs/1708.06997, 2017.

[18] Ž. Emersic, V. Štruc, P. Peer. Ear Recognition: More Than a Survey. *Neurocomputing*, 255:26–39, 2017.