Deep Tattoo Recognition

Xing Di and Vishal M. Patel
Department of Electrical and Computer Engineering
Rutgers, The State University of New Jersey
508 CoRE, 94 Brett Road, Piscataway, NJ 08854
xd55@scarletmail.rutgers.edu, vishal.m.patel@rutgers.edu

Abstract

Tattoo is a soft biometric that indicates discriminative characteristics of a person such as beliefs and personalities. Automatic detection and recognition of tattoo images is a difficult problem. We present deep convolutional neural network-based methods for automatic matching of tattoo images based on the AlexNet and Siamese networks. Furthermore, we show that rather than using a simple contrastive loss function, triplet loss function can significantly improve the performance of a tattoo matching system. Extensive experiments on a recently introduced Tatt-C dataset show that our method is able to capture the meaningful structure of tattoos and performs significantly better than many competitive tattoo recognition algorithms.

1. Introduction

Soft biometrics are physiological and behavioral characteristics that provide some identifying information about an individual [5]. Color of eye, gender, ethnicity, skin color, height, weight, hair color, scar, birthmarks, and tattoos are examples of soft biometrics. Several techniques have been proposed to identify or verify an individual based on soft biometrics [5], [15], [1], [19] in the literature. In particular, person identification and retrieval systems based on tattoos have gained a lot of interest in recent years [11], [7], [8], [12]. Tattoos, in some extent, indicate one's personal beliefs and characteristics. Hence, the analysis of tattoos can lead to a better understanding of one's background and membership to gang and hate groups [11]. They have been used to assist law enforcement in investigations leading to the identification of criminals [13].

In order to promote research and development in tattoo-based recognition applications, a tattoo dataset called Tattoo Recognition Technology - Challenge (Tatt-C) was recently developed by NIST [12], [13]. This dataset contains a total of 16,716 tattoo images collected operationally by law enforcement and is partitioned into five use cases derived from operational scenarios. These use cases are as follows

Figure 1. Samples images from the Tatt-C database. 1st row: images corresponding to the tattoo detection use case., 2nd row: images corresponding to the tattoo similarity use case. 3rd row: images corresponding to the mixed media use case.

- Tattoo Identification: matching different instances of the same tattoo image from the same subject over time,
- Region of Interest: matching a subregion of interest that is contained in a larger tattoo image,
- Mixed Media: matching visually similar or related tattoos using different types of images (i.e., sketches, scanned print, computer graphics, and graffiti),
- Tattoo Similarity: matching visually similar or related tattoos from different subjects,
- Tattoo Detection: detecting whether an image contains a tattoo or not.

In this paper, we mainly focus on the following three use cases - tattoo detection, tattoo similarity and mixed media. Figure 1 shows samples images from the Tatt-C dataset corresponding to these use cases. Tattoo detection has several implications in database maintenance and construction

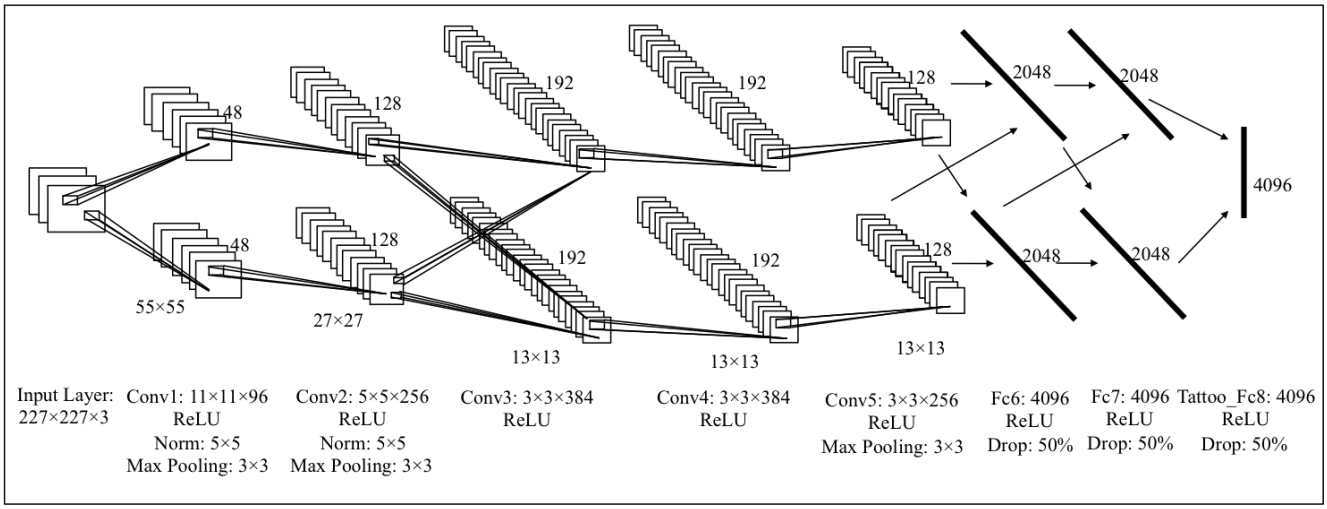

Figure 2. The AlexNet architecture.

when a dataset consists of weakly labeled data. Partially or ambiguously labeled data often makes the automatic interpretation and extraction of different types of images challenging. As was indicated in [13], in the ANSI/NIST Type 10 record, facial mugshot images and scar, mark, tattoo images are stored in the same record type. If some data are mislabeled or unlabeled, then automatic extraction of the data based on image content becomes a major issue.

Tattoo similarity is another use case that has applications in group or gang affiliations. Since members of a group or gang tend to have similar tattoos, one can try to identify individuals belonging to the same gang by looking for people with similar tattoos. In this use case, the objective is to match a probe image with one or more gallery images.

Mixed media is the use case that has application in investigative intelligence gathering where the tattoo is not necessarily captured by a camera but described as a sketch. In this test case, data consists of mixed media and tattoo images and given a mixed media probe image, one has to match one or more tattoos in the dataset [13].

From the use cases described above, we can see that tattoo detection is a two class classification problem and the other two cases, mixed media and tattoo similarity, are both one-to-many verification problems. Previous approaches essentially tackle these problems by first extracting some sort of generative or discriminative features from the given images and then training discriminative classifiers for matching. The performance of these methods is limited by the strength of the features they use. In previous approaches, the features used are often hand-crafted such as Gabor, LBP or SIFT [11], [13]. In recent years, features obtained using deep convolutional neural networks (CNNs) have yielded impressive results on various computer vision applications such as object detection [6], [14], [17] and recognition [10], [3]. Recent studies have shown that in the absence of massive datasets, transfer learning can be effective as it allows one to introduce deep networks without having to train them from scratch [22]. For instance, one can use deep CNNs such as AlexNet [10] or Siamese network [2], [4] pre trained with a large generic dataset such as ImageNet [16] as meaningful feature extractors.

In this paper, we study the performance of deep CNN features on tattoo recognition problems. For the classification problems, such as tattoo detection, we extract fine-tuned deep features based on the AlexNet network using the tattoo images from the Tatt-C dataset and train a linear SVM for classification. For the verification problems, we extract deep features using the Siamese network and match the data using the Euclidean distance as well as a measure based on a triplet loss function.

Rest of the paper is organized as follows. Details of our deep CNN-based methods for tattoo recognition are given in Section 2. Experimental results on the Tatt-C dataset are presented in Section 3. Finally, Section 4 concludes the paper with a brief summary and discussion.

2. Proposed Method

In this section, we describe the details of our proposed methods for tattoo recognition based on AlexNet and Siamese networks.

2.1. Deep Tattoo Detection

The proposed tattoo detection framework consists of two main stages. In the first stage, we extract the deep features based on the AlexNet framework. Figure 2 shows the AlexNet architecture. Then, in the second stage, we train a linear SVM to determine whether a given image contains tattoo or not. We implemented the deep CNN model using caffe [9]. As the AlexNet has been trained on the ImageNet

Name	Type	Filter Size/Stride	Output Size
Conv1	Convolution	$11 \times 11/4$	$55 \times 55 \times 96$
Relu1	ReLU		$55 \times 55 \times 96$
Norm1	LRN	5×5	$55 \times 55 \times 96$
Pool1	Max Pooling	$3 \times 3/2$	$27 \times 27 \times 96$
Conv2	Convolution	$5 \times 5(pad2)/1$	$27 \times 27 \times 256$
Relu2	ReLU		$27 \times 27 \times 256$
Norm2	LRN	5×5	$27 \times 27 \times 256$
Pool2	Max Pooling	$3 \times 3/2$	$13 \times 13 \times 256$
Conv3	Convolution	$3 \times 3(pad1)/1$	$13 \times 13 \times 384$
Relu3	ReLU		$13 \times 13 \times 38)$
Conv4	Convolution	$3 \times 3(pad1)/1$	$13 \times 13 \times 384$
Relu4	ReLU		$13 \times 13 \times 384$
Conv5	Convolution	$3 \times 3(pad1)/1$	$13 \times 13 \times 256$
Relu5	ReLU		$13 \times 13 \times 256$
Pool5	max Pooling	$3 \times 3/2$	$6 \times 6 \times 256$
Fc6	fully connection		4096×1
Relu6	ReLU		4096×1
Drop6	Dropout	50%	4096×1
Fc7	fully connection		4096×1
Relu7	ReLU		4096×1
Fc8_tattoo	fully connection		2×1

Table 1. The AlexNet architecture used in this paper.

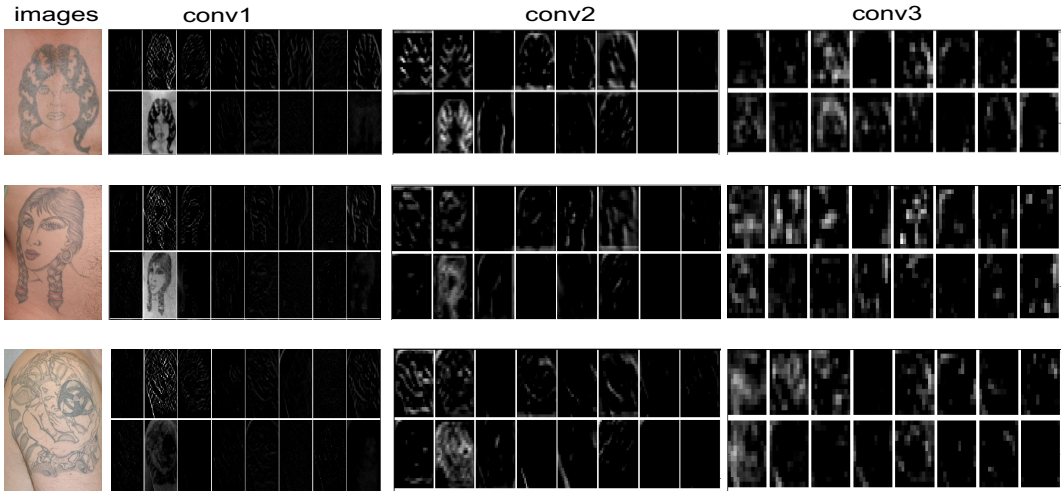

Figure 3. Some feature maps from Conv1, Conv2, and Conv3 layers. The upper feature maps are more robust the illumination changes.

LSVRC-2010 database [16], we fine-tune the network on the Tatt-C dataset for tattoo detection [12].

Table 1 gives the details of the deep CNN architecture used for tattoo detection. All the images, during the training process are scaled into $[0, 1]$ and subtracted from the their mean value. These training images are also flipped about the horizontal and vertical axis before feeding them into the network in order to increase the number of training data for learning the network parameters. During the training and validation phases, we cropped the image into the standard 227×227 size. The basic learning rate for the tattoo detection was set equal to 10^{-4}. The decay rate, gamma, was selected to be 0.1 for every 3500 iterations. The multiplications of the convolutional layers are 1 for weights and 2 for biases. The weights values for the filter and are set randomly according to a Gaussian distribution with 0.01 stan-

dard deviation and weight for the bias is set equal to 1. We set the negative slope to 0 in ReLU layer. The softmax loss layer computes the multinomial logistic loss of the softmax of its inputs. It is conceptually identical to a softmax layer followed by a multinomial logistic loss layer, but provides a more numerically stable gradient [9]. The momentum and total iteration numbers are set equal to 0.9 and 10500, respectively.

After fine-tuning the AlexNet on the tattoo database, we extract the deep feature as the output of the $fc7$ layer, which is a 4096 dimension vector. Then, we implemented a 2-class linear SVM using vlfeat [20] to classify the probe images based on their deep features. The parameter lambda is set equal to 0.01, and the maximum number of iterations is set equal to 10^4.

The tattoo detection dataset has 2349 images, which includes the tattoo and non-tattoo images. Also, there is a ground_truth.txt file, which gives the labels "tattoo" or "non-tattoo" for each image. In this use case, we use label 1 to indicate the tattoo images, and label -1 to indicate the non-tattoo images. Following the standard protocol defined in [13], we use four out of five probe images for training and use the remaining images for testing. For instance, when testing on the 1st probe-list images, we use the images from the 2nd, 3rd, 4th, and 5th probe-lists for training. We repeat this process for all the probe-list images. Figure 3 shows the output from the first three convolutional layers corresponding to three sample images in the Tatt-C dataset. We can see that these features do capture meaningful information about tattoos such as edges, lines and corner points.

2.2. Deep Tattoo Recognition

For the tattoo verification cases such as tattoo similarity and tattoo mixed media use cases, we trained the Siamese network directly on the Tatt-C dataset. The Siamese network used in this paper is shown in Figure 4 and details are given in Table 2. As before, we use the data augmentation by flipping the mixed media and tattoo similarity images horizontally and vertically and scaled them into $[0, 1]$.

Name	Type	Filter Size/Stride	Output Size
Conv1	Convolution	$5 \times 5/1$	$52 \times 42 \times 20$
Pool1	Pooling	$2 \times 2/2$	$26 \times 21 \times 20$
Conv2	Convolution	$5 \times 5/1$	$22 \times 17 \times 50$
Pool2	Pooling	$2 \times 2/2$	$11 \times 9 \times 50$
ip1	InnerProduct		500×1
relu1	ReLU		
ip2	InnerProduct		10×1
feat	InnerProduct		2×1

Table 2. Details of the Siamese network used in this paper for tattoo recognition.

For the mixed media use case, we use the contrastive loss function [4] which is defined as

$$\begin{aligned} L(W) &= \sum_{i=1}^{P} L(W, (Y, X_1, X_2)^i), \\ L(W, (Y, X_1, X_2)^i) &= (1-Y) L_G(E_W(X_1, X_2)^i) \\ &\quad + Y L_I(E_W(X_1, X_2)^i), \end{aligned} \tag{1}$$

where $(Y, X_1, X_2)^i$ is the i-th sample which is composed of a pair of images (X_1, X_2) and a label Y, L_G is the partial loss function for a genuine pair, L_I the partial loss function for an impostor pair, and P is the number of training samples. In caffe, we use the Euclidean distance for $E_W(X_1, X_2)$. The margin we set in the training is 1. The total training iteration is set equal to 7×10^4. The initial learning rate is set equal to 10^{-4} and it decreases by 10% every 2×10^4 iterations. The multiplication learning rate for the neuron is set equal to 1 and 2 for the bias.

There are a total of 453 images (181 probe and 272 gallery) in the mixed media dataset. We also made the "genuine pairwise", which consists of the probe images and their verified gallery images, and the "impostor pairwise", which consists of the probe images and their unverified images. The number of "impostor pairwise" images were much larger than the "genuine pairwise" images. As a result, we randomly chose the equal number of "impostor pairwise" images and "genuine pairwise" images as the training subset. We cropped the images to 56×46. After training the network, output from the "ip2" layer is used as features. Finally, the images are verified based on the Euclidean distances.

For the tattoo similarity use case, rather than using the contrastive loss function, we replace it with the triplet loss function [21], [18]. The triplet loss function is defined as

$$\begin{aligned} L = \sum_{i=1}^{N} \max(0, ||f(x_i^a) - f(x_i^p)||_2^2 \\ - ||f(x_i^a) - f(x_i^n)||_2^2 + \alpha), \end{aligned} \tag{2}$$

where x_i^a is the reference image, x_i^p is the "genuine pairwise" image (positive pairwise), and x_i^n is the "impostor pairwise" (negative pairwise). The threshold α is referred to as "margin". In tattoo similarity case, we replace the contrastive loss function with the triplet loss function. We set the margin equal to 0.005 and the total iteration number to 4×10^4. All the parameters are the same as the original Siamese Network Configuration shown in Table 2 except that the dimension of "ip2" is 256 instead of 10. The initial learning rate is set equal to 0.0002 and decreases to 10% every 1×10^4 iterations. As before, the multiplication learning rates for the neuron is set equal to 1 and 2 for the bias. Tattoo similarity dataset has 2212 images, which consists of 851 probe images and 1361 gallery images. All the images

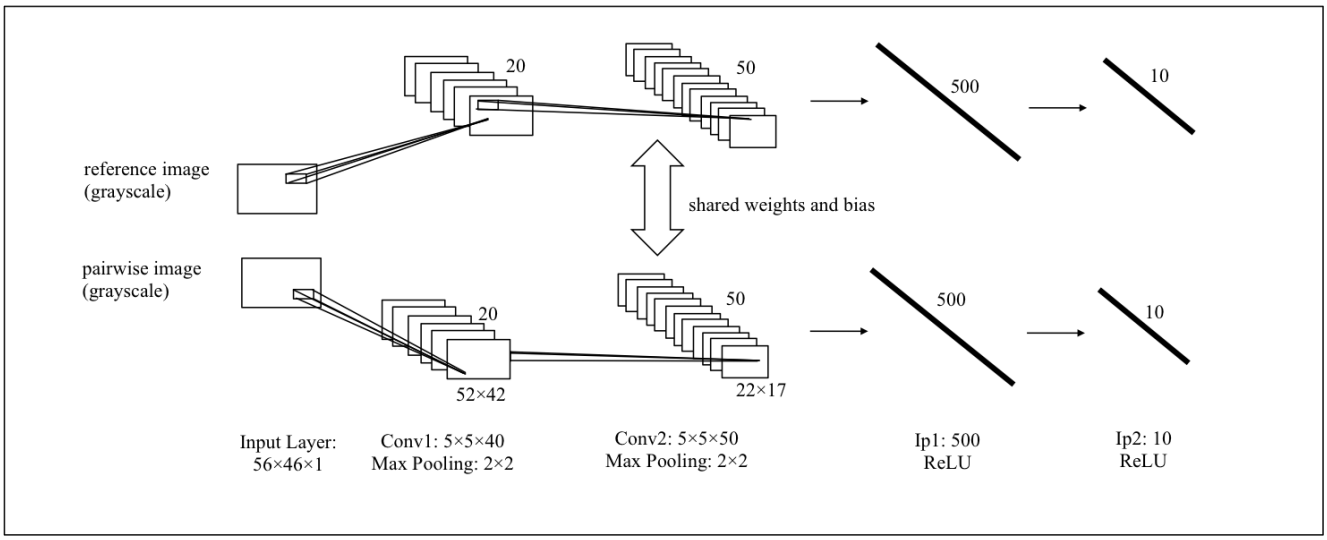

Figure 4. The Siamese network architecture.

for mixed media and tattoo similarity are gray-scaled before training. Output from the "ip2" layer is used as features.

3. Experimental Results

In this section, we present results of our deep CNN-based methods for tattoo recognition on the Tatt-C dataset. We compare the performance of our methods with those reported in [13]. The performance of different methods are compared using accuracy and Cumulative Match Characteristic (CMC) curves. Tattoo accuracy is defined as the number of correctly classified tattoo images TT, divided by the total number of tattoo images N_{tattoo} as

$$Tattoo\ accuracy = \frac{TT}{N_{tattoo}}. \quad (3)$$

Non-tattoo accuracy is defined as the number of correctly classified non-tattoo images NT, divided by the total number of non-tattoo images $N_{non-tattoo}$ as

$$Non-Tattoo\ accuracy = \frac{NT}{N_{non-tattoo}}. \quad (4)$$

The overall accuracy is defined as the sum of correctly classified tattoo and non-tattoo images divided by the total number of images

$$Overall\ accuracy = \frac{TT + NT}{N_{tattoo} + N_{non-tattoo}}. \quad (5)$$

The CMC is defined as the fraction of searches that return the relevant images as a function of the candidate list length. The longer the candidate list, the greater the probability that relevant images are on the list. For searches that have multiple relevant matches in the gallery, the cumulative accuracy or hit rate at any particular rank is calculated with the best-ranked match and represents a best-case scenario.

3.1. Tattoo Detection

The first row of Figure 1 shows some sample images from the Tatt-C dataset corresponding to the detection use case. There are in total 2349 images in this subset - 1349 tattoo images and 1000 non-tattoo images. The non-tattoo images are essentially face images extracted from the Multiple Encounter Database 2 (MEDS- II) [13]. The performance of different methods on the tattoo detection experiment is shown in Table 3. As can be seen from this table, our method outperforms the previous best methods reported in [13] and achieves the overall accuracy of 99.83%.

Figure 5. Wrongly classified images. Only 4 out of 2349 images are wrongly classified by our deep CNN-based method.

In Figure 5, we display the images on which our algorithm fails to correctly detect a tattoo image. In particular, only 4 out of 2349 images are misclassified by our algorithm. Two tattoo images are classified as non-tattoo images and two non-tattoo images are classified as tattoo images. In

Algorithm	Non-Tattoo Accuracy	Tattoo Accuracy	Overall Accuracy
CEA_1	0.988	0.932	0.956
Compass	0.386	0.798	0.622
MITRE_1	0.750	0.734	0.741
MITRE_2	0.948	0.924	0.934
MorphoTrak	0.950	0.972	0.963
Deep Tattoo	0.9980	0.9985	0.9983

Table 3. Performance comparison of different methods on the detection use case. Results other than Deep Tattoo are taken directly from [13].

the first row of this figure, a tattoo image is recognized as a face image and in the second row, two face images are recognized as tattoo images. As can be seen from this figure, the reason why our method fails is that the wrongly classified tattoo image is a face-like image and our algorithm classifies it as a face image.

3.2. Mixed Media

Results of different methods corresponding to the mixed media use case are shown in Table 4 and in Figure 6. As can be seen from this table, our method significantly outperforms the previous methods and achieves 100% accuracy at rank 28. The descriptor used by MITRE is the shape contexts-based and Compass uses some low-level features like color, brightness, contrast, etc. In contrast, our method uses deep features directly learned on the tattoo images. As a result, our method is able to capture the salient information that is present in the tattoo images better than the other methods. This experiment clearly shows the significance of deep features compared to hand-crafted features for tattoo recognition.

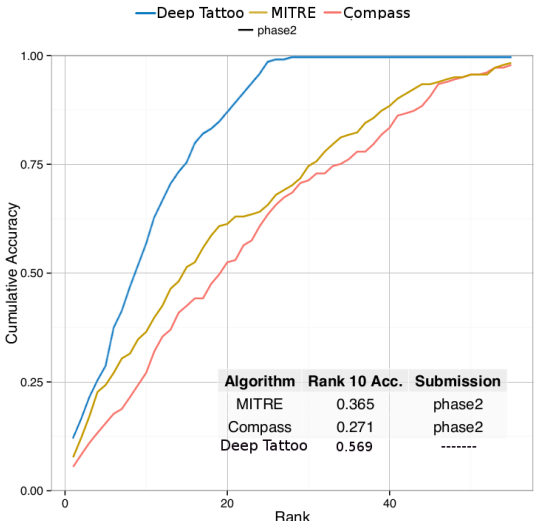

Figure 6. The CMC curves corresponding to different methods on the mixed media use case.

To gain further insight into our method, in Figure 7 we show some correctly matched and wrongly matched samples. First row displays images that are correctly classified and the second row displays images on which our method fails to correctly classify the mixed media images. Again the reason why our method correctly classifies mixed media images as tattoo images is because they look very similar to the tattoo images. This can be clearly seen by comparing images shown in the second row of Figure 7.

Figure 7. Sample result from the mixed media use case. First row: correct matching. Second row: failed matches.

3.3. Tattoo Similarity

Table 5 and Figure 8 show the results of different methods on the tattoo similarity use case. As can be seen from these results, our method outperforms the previous methods especially when the triplet loss function incorporated within our framework. For instance, at rank-10, our method with triplet loss function gives an accuracy of 16.40% compared to 14.9%, 7.4% and 11.1% for MITRE, Compass, and non-triplet based method. Again, this experiment clearly shows that one can significantly improve the performance of a tattoo recognition algorithm by using deep features.

In Figure 9, we display a few correctly matched and incorrectly matched images for the tattoo similarity use case. First row of this figure shows the correctly matched images and the second row shows the incorrectly matched images.

Algorithm	Submission	Rank 1 Accuracy	Rank 10 Accuracy	Rank 20 Accuracy	Rank 30 Accuracy
Compass	phase2	0.055	0.271	0.525	0.713
MITRE	phase2	0.077	0.365	0.613	0.746
Deep Tattoo	- - - - -	0.122	0.569	0.873	1

Table 4. Performance comparison of different methods on the mixed media use case. Results other than Deep Tattoo are taken directly from [13]. Number of probes: 181, Average gallery size: 55.

Algorithm	Submission	Rank 1 Accuracy	Rank 10 Accuracy	Rank 20 Accuracy	Rank 30 Accuracy
Compass	phase2	0.005	0.074	0.147	0.199
MITRE	phase2	0.035	0.149	0.239	0.309
Deep Tattoo	triplet	0.055	0.164	0.249	0.316
Deep Tattoo	non-triplet	0.017	0.111	0.155	0.210

Table 5. Performance comparison of different methods on the tattoo similarity media use case. Results other than Deep Tattoo are taken directly from [13]. Number of probes: 851, Average gallery size: 272.

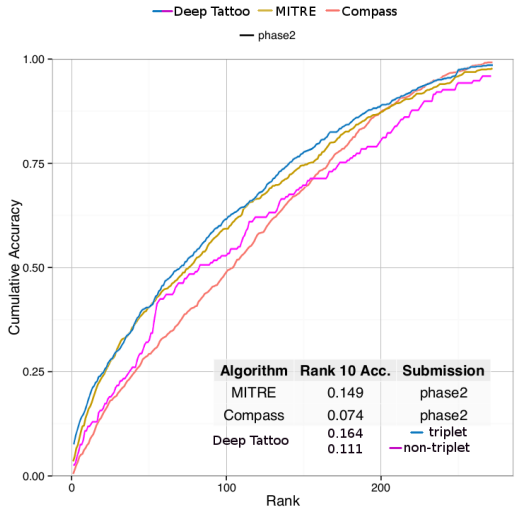

Figure 8. The CMC curves corresponding to different methods on the tattoo similarity use case.

As can be seen from this figure, these images are extremely difficult to match as they contain various illumination, pose and resolution variations. One of the reasons why our deep feature-based method does not work well in this particular use case is that we do not have a significant number of tattoo images with different variations to train our deep models.

4. Conclusion

In this paper, we presented deep feature-based methods for tattoo detection and recognition using the recently intruded AlexNet and Siamese networks. Furthermore, we showed that rather than using a simple contrastive loss function, triplet loss function can significantly improve the performance of a tattoo matching system based on deep features. Extensive experiments on the Tatt-C dataset demonstrated the effectiveness of our proposed approach.

Figure 9. Sample result from the mixed media use case. First row: 629 correct matching. Second row: failed matches.

References

[1] A.K.Jain and U.Park. Facial marks: soft biometric for face recognition. In *IEEE International Conference on Image Processing*, pages 37–40, November 2009. 1

[2] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah. Signature verification using a “siamese" time delay neural network. In J. D. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems 6*, pages 737–744. Morgan-Kaufmann, 1994. 2

[3] J.-C. Chen, V. M. Patel, and R. Chellappa. Unconstrained face verification using deep cnn features. In *IEEE Winter conference on Applications of Computer Vision*, 2016. 2

[4] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 539–546. IEEE, 2005. 2, 4

[5] A. Dantcheva, P. Elia, and A. Ross. What else does your biometric data reveal? a survey on soft biometrics. *IEEE Transactions on Information Forensics and Security*, 11(3):441–467, March 2016. 1

[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Computer Vision and Pattern Recognition*, 2014. 2

[7] B. Heflin, W. Scheirer, and T. Boult. Detecting and classifying scars, marks, and tattoos found in the wild. In *International Conference on Biometrics: Theory, Applications and Systems*, pages 31–38, Sept 2012. 1

[8] A. Jain, J.-E. Lee, and R. Jin. Tattoo-id: Automatic tattoo image retrieval for suspect and victim identification. In H.-S. Ip, O. Au, H. Leung, M.-T. Sun, W.-Y. Ma, and S.-M. Hu, editors, *Advances in Multimedia Information Processing*, volume 4810 of *Lecture Notes in Computer Science*, pages 256–265. Springer Berlin Heidelberg, 2007. 1

[9] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014. 2, 4

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. pages 1097–1105, 2012. 2

[11] J.-E. Lee, R. Jin, A. Jain, and W. Tong. Image retrieval in forensics: Tattoo image database application. *IEEE MultiMedia*, 19(1):40–49, Jan 2012. 1, 2

[12] M. Ngan and P. Grother. Tattoo recognition technology - challenge (Tatt-C): an open tattoo database for developing tattoo recognition research. In *IEEE International Conference on Identity, Security and Behavior Analysis*, pages 1–6, March 2015. 1, 3

[13] M. Ngan, G. W. Quinn, and P. Grother. Tattoo recognition technology–challenge (tatt-c) outcomes and recommendations. Technical Report NISTIR 8078, National Institute of Standards and Technology, Sept. 2015. 1, 2, 4, 5, 6, 7

[14] R. Ranjan, V. M. Patel, and R. Chellappa. A deep pyramid deformable part model for face detection. In *IEEE International Conference on Biometrics Theory, Applications and Systems*, pages 1–8, Sept 2015. 2

[15] D. Reid, S. Samangooei, C. Chen, M. Nixon, and A. Ross. Soft biometrics for surveillance: An overview. *Handbook of Statistics*, 31:327–352, 2013. 1

[16] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 2, 3

[17] S. Sarkar, V. M. Patel, and R. Chellappa. Deep feature-based face detection on mobile devices. In *IEEE International Conference on Identity, Security and Behavior Analysis*, 2016. 2

[18] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. *CoRR*, abs/1503.03832, 2015. 4

[19] P. Tome, J. Fierrez, R. Vera-Rodriguez, and M. Nixon. Soft biometrics and their application in person recognition at a distance. *IEEE Transactions on Information Forensics and Security*, 9(3):464–475, March 2014. 1

[20] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org/, 2008. 4

[21] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning fine-grained image similarity with deep ranking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1386–1393, 2014. 4

[22] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? *CoRR*, abs/1411.1792, 2014. 2