Minutiae-Informed Contrastive Learning for Fingermark Recognition

Lara Anžur
Faculty of Computer and Information Science
University of Ljubljana
Ljubljana, Slovenia
la4710@student.uni-lj.si

Peter Peer
Faculty of Computer and Information Science
University of Ljubljana
Ljubljana, Slovenia
peter.peer@fri.uni-lj.si

Tim Oblak
Faculty of Computer and Information Science
University of Ljubljana
Ljubljana, Slovenia
tim.oblak@fri.uni-lj.si

***Abstract*—Automated fingermark identification, a critical aspect of forensic investigations, is often challenged by low-quality and partial samples. To address some of these challenges, we participated in the IJCB 2024 Latent in the Wild Fingerprint Recognition Competition, where our solution achieved third place. This paper presents several enhancements to our original approach, including the integration of a Spatial Transformer Network and a U-Net-based minutiae predictor. Combined with a contrastive learning strategy, these improvements enable the generation of highly representative feature vectors capable of handling complex fingermarks. The revised solution demonstrates a significant performance improvement, indicating that it would surpass our original submission and achieve a higher placement in the IJCB competition.**

***Index Terms*—latent fingerprints, fingermarks, forensics, biometrics, identification, verification, minutiae**

I. Introduction

Fingerprint identification is to this day one of the most commonly used methods to identify suspects in forensic investigations. The unique ridge patterns on our fingertips carry significant informational value, making them highly effective for identification purposes. However, the identification of fingermarks (latent fingerprints), which are often partial, degraded, or distorted impressions, still presents significant challenge to automated identification systems.

Traditionally, fingermark identification relied on manual and semi-automated techniques, by leveraging expert knowledge and heuristic algorithms to extract minutiae and other friction ridge features. The emergence of deep learning enabled the automation of feature extraction, however, critical questions remain about how to best guide the learning procedure. Challenges include selecting optimal learning strategy, incorporating domain knowledge, and refining the optimization process to effectively capture the unique characteristics of fingermarks. Development of recognition systems also goes hand-in-hand with work on utility-based quality assessment [1], [2].

This paper proposes an improvement of our existing approach, which achieved third place at the IJCB 2024 Latent in the Wild Fingerprint Recognition Competition [3]. Despite securing third place, our approach was notably outperformed by the other competitors, both of which were submissions from commercial providers. We now introduce several improvements to the original design by including a Spatial Transformer Network (STN) and incorporating an intermediate minutiae predictor. The improved approach produces a much more representative feature vector, which is able to capture the properties of even the more complex fingermarks.

To address the challenges of fingermark identification, we make the following contributions in this paper:

- **Minutiae probability prediction**: We train a U-Net model to predict the probability of minutiae points in the input image. This probabilistic encoding of minutiae locations is particularly effective for fingermarks with ambiguous or incomplete features.
- **Rotation invariant feature vector**: We address the challenge of rotated fingermarks. By incorporating a Spatial Transformer Network (STN), the model is able to learn rotation transformations, which results in more rotation-invariant fingermark representations.
- **Contrastive learning framework**: We employ a contrastive learning approach using triplet loss to learn robust representations and construct a latent space based on unique fingerprint properties.

II. Related Work

The field of friction has progressed significantly, with techniques being developed for fingerprint and fingermark recognition. Since there is a notable overlap between these two areas, we rather categorize the existing work by distinguishing between traditional and deep learning approaches.

Traditional Approaches. Notable methods in this area include NIST's NBIS [4], which offers several algorithms that can be used to build an recognition pipeline. The methods include `mindtct` for detecting and extracting minutiae points, and `bozorth3`, which is a minutiae-based matcher. The more recent OpenAFQA framework [5] offers a collection of open-source algorithms for fingerprint and fingermark processing. Other known techniques focus on combining both local and global minutiae information [6]. Some approaches also incorporate additional fingerprint features, such as singularities and ridge maps, to improve representation [7], [8]. Additionally, minutiae clustering methods have been investigated [9], [10].

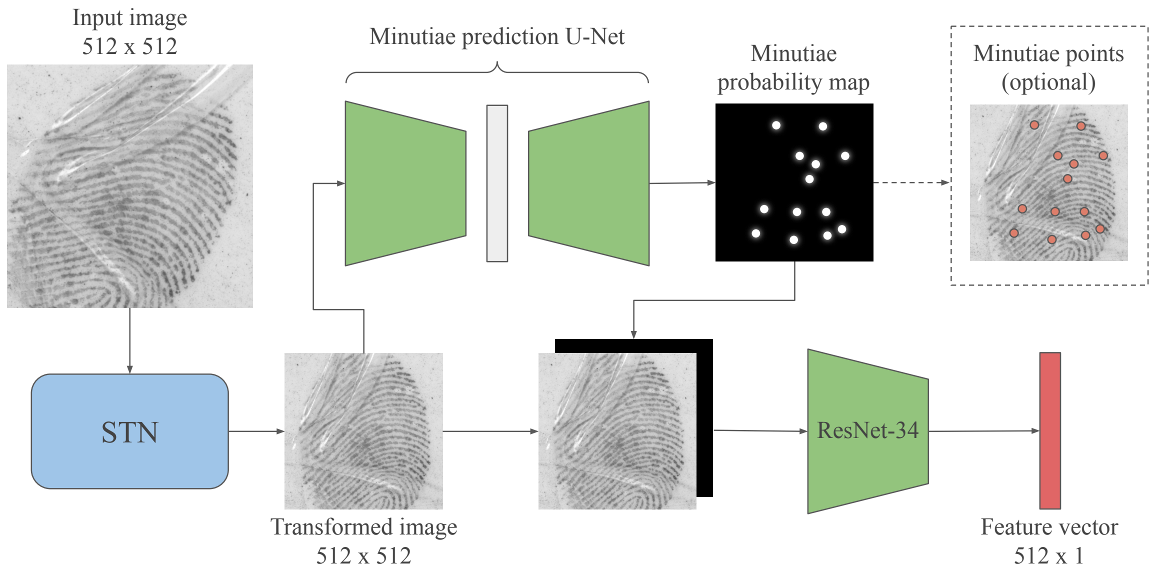

Fig. 1. Our proposed method for fingermark identification. The input image is first processed by a Spatial Transformer Network (STN) module, followed by minutiae detection using U-Net. The transformed image and the minutiae probability map are then concatenated and fed into a ResNet-34 encoder, which generates the final representation of the input image.

Deep Learning Approaches. Deep learning has automated feature extraction in fingerprint recognition, but it is often combined with domain knowledge to address challenges posed by poor-quality fingerprints. To overcome the limitation of CNNs lacking rotational invariance, alignment algorithms such as spatial transformer networks (STN) [11] and orientation field dictionaries [12] are employed. Patch embeddings are frequently used to extract minutiae from input images [13], [14], while virtual minutiae are introduced in some methods to enhance robustness when dealing with low-quality fingermarks [15], [16]. MSU Latent AFIS [17], [18] incorporates three complementary minutiae detection methods and a fourth texture-based technique, specifically designed to handle cases with small images or insufficient minutiae.

III. Proposed Method

In this chapter, we outline the structure of our proposed method, as illustrated in Fig. 1. The architecture consists of three main modules: the Spatial Transformer Network (STN), a ResNet-34 encoder, and a U-Net minutiae probability predictor.

A. *Contrastive learning using Siamese Neural Networks*

Forensic laboratories often deal with large open-set datasets, where the number of possible identities are not known in advance. To account for this, we opted to train our model in a Siamese Network configuration [19], [20]. During training, two or more instances of the same architecture use shared weights to predict features for multiple images. A specialized loss functions is then used to update the shared weights based on the similarity of output vectors.

Loss Function. To facilitate contrastive learning, we use triplet loss as our loss function. In triplet loss, the model receives triplets of images as input: one serving as the anchor, the second as a positive example (similar to the anchor), and the third as a negative example (dissimilar to the anchor). This approach helps the model learn to better distinguish between similar and dissimilar identities. Since we process a triplet of images at once, we obtain three feature vectors. The distances between these vectors are then calculated to compute the loss, with the objective of maximizing the distance between the negative example and the anchor, while minimizing the distance between the positive example and the anchor.

Triplet selection. To effectively train the model, the selection of triplets is crucial, as we aim to construct challenging triplets that contribute to more effective learning. Our approach involved randomly selecting two distinct identities from the dataset (the anchor and positive example belonging to the same identity, while the negative example came from a different identity). For the anchor and negative examples, we randomly selected samples within each identity, regardless of whether they represented fingermarks or fingerprints. However, for positive examples, we excluded reference fingerprints from the selection process to avoid creating "easy triplets". If the anchor and positive examples were chosen as different instances of the same fingerprint, they would likely be more similar to each other than to the negative example, resulting in an easy triplet that could degrade the performance of the model.

B. *Spatial Transformer Network*

As CNNs are not spatially invariant to input data, we address this limitation by introducing a Spatial Transformer Network (STN) [21] to our existing approach. We incorporate it as a trainable module into the network architecture, right after the input. This allows the network to learn spatial and rotational invariance dynamically, improving recognition without prior training or supervision.

An STN comprises three main components: the localization network, the grid generator, and the sampler. The process begins with the localization network, which predicts transformation parameters θ for an affine transformation using regression. These parameters are specific to each input image and guide the spatial transformation. The grid generator uses θ to compute a sampling grid that maps transformed image coordinates to the original image space. Finally, the sampler applies bilinear interpolation to produce the transformed output image based on the original image and the sampling grid.

C. Predicting minutiae probability

We integrate domain knowledge as one of the additional modules. A U-Net-based model is introduced to predict minutiae locations from input images, generating a probability matrix.

Minutiae Probability Map. For minutiae probability prediction, we adopt the U-Net architecture [22], which is widely used for semantic segmentation tasks. It consists of an encoder and a decoder connected by a bridge, with skip connections between corresponding layers.

The model is designed to output a probability value x $\in$ [0,1] for each pixel, representing the likelihood of a minutia at that location. To achieve this, we apply a sigmoid activation function at the final layer. We trained the U-Net on fingermark images, annotated manually by forensic examiners. Specifically, we used the minutiae annotations provided by NIST in the SD 302 dataset. The weights of the U-Net are then frozen during training of the SNN.

Minutiae detection. By default, the predicted minutiae probability map is only used internally by the recognition model, however, it can also be used to extract a corresponding list of minutiae points. To identify the location of individual minutiae, local maxima are detected within the probability map. Additionally, the probability value associated with each minutia point can be retrieved. This process is illustrated in Fig. 2.

D. Prediction pipeline

The input image is first processed by an STN and then fed through a U-Net, generating a probability map for minutiae locations. This output is combined with the transformed image to create a two-channel data matrix, which is then fed into the final encoder. The result is a reduced representation of the input image, a normalized feature vector of size 512.

Preprocessing. We preprocess the images before feeding them to our model. Each image is first converted to grayscale, then padded with white pixels to form a square image. The images are then resized to 512×512 pixels for consistency. This allows us to retain the proper aspect ratio without introducing any spatial distortions. By resizing to the same image dimensions, the model also learns the scale of input

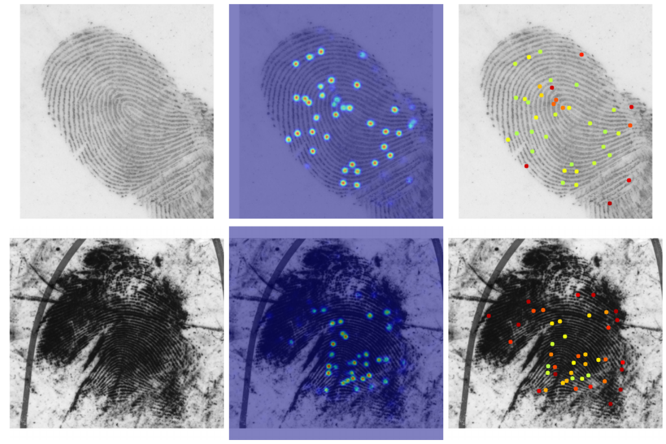

Fig. 2. Intermediate minutiae prediction. By searching for local maxima, the predicted minutiae probability map (middle) can be converted into a list of minutiae points along with their probability values (right).

images, thus making making the method agnostic to image resolution (pixels per inch, PPI).

Image encoder. For the main encoder, we use a Resnet-34 [23] with pretrained weights. Since the model was originally trained on RGB images and our fingermark images are grayscale, we modify the structure by averaging the weights of the first convolution layer from 3 to 1. We also modify the final fully connected layer, which typically outputs class probabilities, to instead produce a feature vector. The output feature vector is normalized in the end, which effectively puts all output vectors on a hypersphere.

Matching. During inference, feature vectors are obtained for the query fingermark image, which is then compared with feature vectors of fingerprints in the reference dataset. The feature vectors are compared using the Euclidean distance d, and the matching score is then computed as $1 - \frac{d}{2}$.

IV. Experiments

In this chapter we first describe our experimental settings and then presents the results of the evaluation using relevant performance metrics.

A. Experimental setting

Datasets. The data utilized in this study consists of a several datasets, including NIST SD 302 [24], NIST SD 27 [25] and the LFIW dataset [26]. In total, we collected 4,640 distinct identities, each represented by multiple samples.

Metrics. We evaluated our models using several metrics. To evaluate the performance of the system in an identification scenario, we report the Rank-25 identification rate. We assess the performance of verification through the genuine-impostor distribution, Equal Error Rate (EER) and the Area Under the Curve (AUC). We also report the Failure To Enroll Rate (FTER), which indicates the percentage of input images rejected during inference due to missing features or other system constraints.

Implementation details. The models were developed using the PyTorch framework. For training, evaluation, and testing

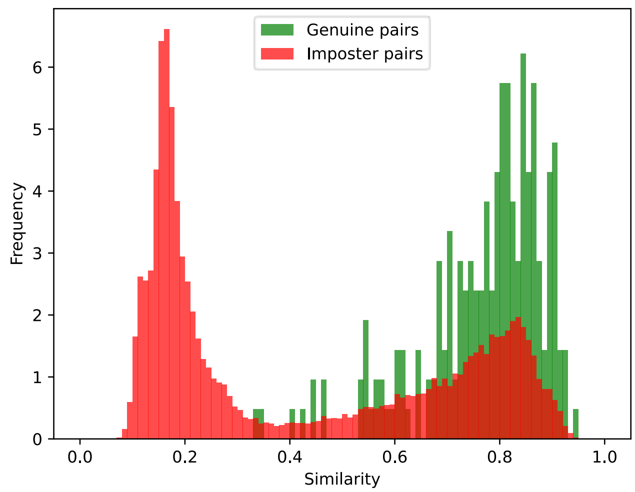

Fig. 3. Genuine and impostor distributions for our proposed method.

purposes, we utilized an NVIDIA GeForce RTX 3080 GPU with CUDA 12.1. Models were trained with batch size of 4 and learning rate of 10^{-5} with Adam optimizer.

B. Results

In this section, we present the results of conducted experiments. The model achieved a Rank-25 identification rate of 41%. In Fig. 3, we present the genuine and impostor distributions for our data. It is visible that our model is generally able to identify genuine pairs, but this comes at a cost of many false positive attributions. This may be due to the large number of poor quality fingermark images in the dataset. Nevertheless, prediction confidence is much higher on the lower end of the matching score range.

	AUC	EER	FTER [%]
MSU-AFIS	0.57	0.48	7.7
MCC	0.75	0.32	36.7
VeriFinger_12.3	0.81	0.27	40.8
VeriFinger_13.1	**0.85**	**0.23**	6.2
LatentMinuComp_v0	0.79	0.29	2.2
Proposed_IJCB	0.55	0.48	**0.0**
Proposed (this paper)	0.82	0.26	**0.0**

TABLE I
COMPARISON OF METRICS AUC, EER AND FTER TO BASELINE COMPETITION MODELS, TOP THREE CONTESTANTS AND OUR NEWLY PROPOSED APPROACH (BOLD = BEST, UNDERLINED = SECOND BEST).

In Tab. I, we present the scores for performance metrics AUC, EER and FTER, and compare our new approach to the results of the IJCB competition. The first three rows are baseline models, second three are the top three competitors (with our contribution **Proposed_IJCB**) and the final row (**Proposed**) represents our approach presented in this paper.

Our newly proposed method significantly outperformed the version we submitted to the competition. It is important to note that the evaluation was made on a test set, which differs from competition's hidden test set, but is sampled from the same data distribution. As such, the results on the hidden test set may differ slightly. Nevertheless, we came close to matching the performance of the competition's winner as we made notable improvements in performance metrics, increasing the AUC by 0.27 and reducing the EER from 0.48 to 0.28. We also remain the only team to achieve an FTER score of 0%.

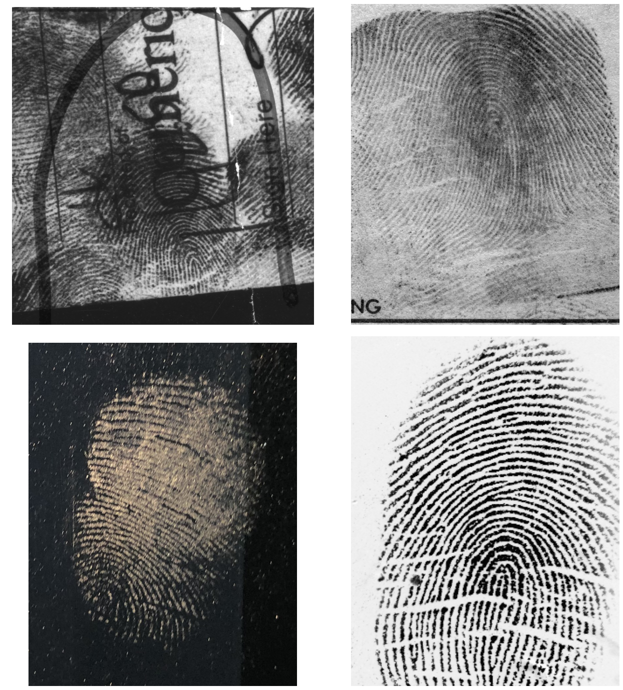

Fig. 4. Qualitative evaluation of our model. In the first row, there is a correctly identified pair at Rank-1, while the second row shows a genuine pair with a low matching score (Rank-20 or higher).

Last, we present two pairs of images side by side for qualitative evaluation of our proposed method in Fig. 4. The first row shows a correctly identified pair at Rank-1, while the second displays a genuine pair, which achieved a low matching score and produced a match outside of the top 25. While both fingermarks are partial impressions, the top fingermark retains visible ridge structure despite the background clutter. On the other hand, the bottom fingermark contains various ambiguous features due to the thickness of individual ridges and general blurriness of the image. This qualitative comparison emphasizes the role of fingerprint quality in the matching process.

V. CONCLUSION

This paper addresses the challenge of automated fingermark identification, integrating a Spatial Transformer Network (STN), a U-Net-based minutiae predictor, and a contrastive learning framework. Our method presents an improvement over our previous approach and closes the gap with the proprietary state-of-the-art methods, resulting in better performance across key metrics, including EER and AUC.

Despite the visible improvements, the final results indicate that there is still considerable room for progress in this area.

Poor input image quality is likely the main factor behind most misidentified genuine pairs, and one approach to address this could be using a quality estimator to filter out low-quality samples. Another option would be to exclude instances with an insufficient number of detected minutiae during evaluation, though at the cost of increasing the FTER. To address the issue of undetected minutiae, alternative datasets containing annotated minutiae could also be explored.

References

[1] T. Oblak, R. Haraksim, P. Peer, and L. Beslay, "Fingermark quality assessment framework with classic and deep learning ensemble models," *Knowledge-Based Systems*, vol. 250, p. 109148, 2022.

[2] T. Oblak, R. Haraksim, L. Beslay, and P. Peer, "Probabilistic fingermark quality assessment with quality region localisation," *Sensors*, vol. 23, no. 8, 2023.

[3] X. Liu, R. Wang, T. Oblak, L. Anžur, P. Peer, M. Lange, D. Stumpf, S. Utcke, E. Borcovas, and K. Raja, "Latent in the wild fingerprint recognition competition," in *Proceedings of the IEEE International Joint Conference on Biometrics*, 2024, sprejeto v objavo.

[4] K. Ko, "User's guide to nist biometric image software (nbis)," National Institute of Standards and Technology, Tech. Rep., 2007.

[5] T. Oblak, R. Haraksim, L. Beslay, and P. Peer, "Fingermark quality assessment: An open-source toolbox," in *International Conference of the Biometrics Special Interest Group (BIOSIG)*, 2021, pp. 1–7.

[6] R. Cappelli, M. Ferrara, and D. Maltoni, "Minutia cylinder-code: A new representation and matching technique for fingerprint recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, pp. 2128–2141, 2010.

[7] A. K. Jain and J. Feng, "Latent fingerprint matching," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 33, pp. 88–100, 2010.

[8] R. P. Krish, J. Fierrez, D. Ramos, F. Alonso-Fernandez, and J. J. Bigun, "Improving automated latent fingerprint identification using extended minutia types," *Information Fusion*, vol. 50, pp. 9–19, 2019.

[9] U. U. Deshpande, V. S. Malemath, S. M. Patil, and S. V. Chaugule, "Automatic latent fingerprint identification system using scale and rotation invariant minutiae features," *International Journal of Information Technology*, vol. 14, pp. 1025–1039, 2022.

[10] M. A. Medina-Pérez, A. M. Moreno, M. A. F. Ballester, M. García-Borroto, O. Loyola-González, and L. Altamirano-Robles, "Latent fingerprint identification using deformable minutiae clustering," *Neurocomputing*, vol. 175, pp. 851–865, 2016.

[20] G. Koch, R. Zemel, R. Salakhutdinov *et al.*, "Siamese neural networks for one-shot image recognition," in *ICML deep learning workshop*, Lille, France, 2015, pp. 1–30.

[11] J. J. Engelsma, K. Cao, and A. K. Jain, "Learning a fixed-length fingerprint representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, pp. 1981–1997, 2021.

[12] K. Cao and A. K. Jain, "Fingerprint indexing and matching: An integrated approach," in *IEEE International Joint Conference on Biometrics (IJCB)*, 2017, pp. 437–445.

[13] H. İ. Öztürk, B. Selbes, and Y. Artan, "Minnet: Minutia patch embedding network for automated latent fingerprint recognition," in *Conference on Computer Vision and Pattern Recognition*, 2022, pp. 1627–1635.

[14] D. L. Nguyen, K. Cao, and A. K. Jain, "Robust minutiae extractor: Integrating deep networks and fingerprint domain knowledge," in *International Conference on Biometrics (ICB)*. IEEE, 2018, pp. 9–16.

[15] S. A. Grosz and A. K. Jain, "Latent fingerprint recognition: Fusion of local and global embeddings," *IEEE Transactions on Information Forensics and Security*, 2023.

[16] K. Cao and A. K. Jain, "Latent fingerprint recognition: role of texture template," in *International conference on biometrics theory, applications and systems (BTAS)*. IEEE, 2018, pp. 1–9.

[17] K. Cao, D. Nguyen, C. Tymoszek, and A. K. Jain, "End-to-end latent fingerprint search," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 880–894, 2019.

[18] K. Cao and A. K. Jain, "Automated latent fingerprint recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 4, pp. 788–800, 2019.

[19] D. Chicco, *Siamese Neural Networks: An Overview*. Springer US, 2021, pp. 73–94.

[21] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," in *Advances in Neural Information Processing Systems*, vol. 28. Curran Associates, Inc., 2015.

[22] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

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

[24] G. Fiumara, P. A. Flanagan, J. D. Grantham, K. Ko, K. Marshall, M. Schwarz, E. Tabassi, B. Woodgate, and C. Boehnen, "Nist special database 302: Nail to nail fingerprint challenge," National Institute of Standards and Technology, Tech. Rep., 2019.

[25] M. G. and R. M. McCabe, "Nist special database 27: Fingerprint minutiae from latent and matching tenprint images," National Institute of Standards and Technology, Tech. Rep., 2000.

[26] X. Liu, K. Raja, R. Wang, H. Qiu, H. Wu, D. Sun, Q. Zheng, N. Liu, X. Wang, G. Huang *et al.*, "A latent fingerprint in the wild database," *IEEE Transactions on Information Forensics and Security*, 2024.