A Study of the Shearing Section Trace Matching Technology Based on Elastic Shape Metric and Deep Learning

Nan Pan,[1*] Yi Liu,[2] Dilin Pan,[2] Xin Shen,[3] and Xin Wang[3]

[1]Faculty of Civil Aviation and Aeronautical, Kunming University of Science & Technology,
No. 727 Jingming South Road, Chenggong University Town, Kunming 650500, P.R. China
[2]Kunming SNLab Technology Co., Ltd.,
Architect Business Center 2202 Room Dong Feng Dong Road 47#, Kunming 650228, P.R. China
[3]Metrology Center of Yunnan Power Grid Co., Ltd., No. 73 Tuodong Road, Kunming 650017, P.R. China

The practical application of shearing linear traces in a crime scene is severely restricted because of complex shapes and markedly high randomness. In this study, a more efficient matching model for clamp cutting surface traces is proposed. To this end, key theories and algorithms are further studied. The overall research approaches are as follows: the isolated forest algorithm is used for the processing of abnormal signals detected, followed by a multiscale registration framework to extract trace curve profiles, and the square speed function optimization elastic shape metric algorithm to map the profiles into an embedding. A parametric shared conjoined triple deep learning model is used, which is suitable for trace features and optimizing the triplet selection and data augmentation strategies. This system is trained by minimizing a triplet loss function, so that a similarity measure is defined by the L^2 distance in this embedding. Finally, the trained model is used for similarity matching for the test set. The matching effect is continuously evaluated through experiments to provide investigators an efficient string detection method for traces.

1. Introduction

Before 2020, China's high-speed rail will reach a total length of 30000 km, covering more than 80% of large cities. At the same time as it will continue to grow, 74 civil aviation airports and 850 general airports will be built. Along railways, there are a large number of optical cables, leaky cables, signal cables, cables for rail transit vehicles, and through-ground lines. There are also a large number of navigation lights and communication optical cables around various airports.[1] The inner conductors of many cables are made of copper or other precious metals, a coveted target for criminals owing to their high economic value. This has caused the failure of the above-mentioned systems, leading to several accidents, significant loss of lives, and diminished property safety.[2]

The examination of crime scenes has showed that criminals use wire cutters, cable cutters, pliers, and other large shear tools to sever cables. Tool traces, which are scratches on the surface of the body, could be caused by the pressure of a linear deformation. Traces from the broken ends are frequently found on the surface at the crime scene.[3–6]

The single-point laser test has the characteristics of small influence of ambient light, high precision, small data file size, good frequency response characteristics, and so forth. It is very effective for noncontact trace measurement.[7] The weight-sharing network structure of the convolutional neural network (CNN) can circumvent the complex feature extraction and data reconstruction processes in the conventional recognition algorithm. It is robust enough to be highly invariant to translation, scaling, tilting, and common-form deformation, which have been widely used in image and speech recognition. Therefore, combining the single-point laser test with CNN and applying it to the similarity matching of clamped-line traces are feasible. However, the following challenges remain in solving practical problems:[8]

1) Trace feature extraction: Signals detected by laser displacement sensors often contain considerable redundant anomaly information owing to mechanical errors and external disturbances. Such abnormal signals should be adaptively detected and corrected to improve the quality and accuracy of subsequent work. Suspects often leave nearly a thousand cable breaks at the crime scene. The same clamp tool has many different shear forces and displacement deviations, and its morphological characteristics will also cause significantly different linear traces. Trace feature extraction has the same immutability for the same tool trace, has high discrimination capability for different tool traces, and requires no manual annotation intervention for mapping the traces into the embedded layer.

2) The model structure, convolution size, feature map, and pooling layer size of the laser detection features that are suitable for linear traces should be determined experimentally. Since CNN cannot understand the physical differences of features, making it easy to apply to invalid features such as detection angle nuances and artificial data enhancement, the training set must contain samples from different categories and of limited complexity and sufficient robustness to allow CNN to identify differences.

3) The diameter of the cable breakage does not exceed 1 cm, only part of the clamp contact surface shape is retained on the bearing body, and two trace detection signals are likely to have only a partial overlap. Under this condition, the computational complexity of two discrete sequence point-to-point similarity metrics is very large ($N \times N$), and it is necessary to select a similarity calculation method that can accurately distinguish individual differences between traces without causing a surge in computational complexity. At the same time, in order to accurately assess how relevant linear traces are discovered, it is necessary to develop a more stable and accurate matching sorting standard.

To solve the above problems, a linear trace intelligent matching method based on deep learning is designed. The method uses a multiscale registration framework to extract the outline of the trace coarse-transition curve and introduces the square-speed function to optimize the elastic shape measurement algorithm to complete the curve profile embedding layer mapping. A trace feature parameter sharing joint triple deep learning model is developed, with optimized triplet selection and data enhancement strategies, using the L^2 distance for embedded layer

similarity calculation. The practicality and effectiveness of the proposed algorithm are verified by experiments involving the use of actual mark inference cable cutting tools.

2. Trace Curve Profile Extraction and Embedded Layer Mapping

2.1 Abnormal data processing

Abnormal data, which is data that is significantly different from the surrounding data, is mostly created by excessive reflection or vibration. A signal obtained by single-point laser detection signal is used as the test data input on the basis of the isolated forest algorithm, with subsampling on the training set to build multiple *iTree* and form an *iForest*. The anomaly index of all samples is calculated, and then the K-means clustering of all one-dimensional data anomaly indices is performed to estimate the current difference threshold. Abnormal samples below this threshold are rejected and corrected by exponentially weighted moving averages based on near-normal data.

2.2 Curve profile fast extraction

A multiscale registration strategy is used to register the profiles of the trace signals formed by different tools and the same tool at different angles (Fig. 1). Registration is carried out in multiple steps with increasingly greater detail. Starting with a coarse structure, only structures larger than a given wavelength λ are considered for each scale. In each step, optimal translation and scaling parameters are determined and used for the next initialization, and the parameters formed in the final step are ultimately used to transform the entire profile including all structures to produce the final registration result.
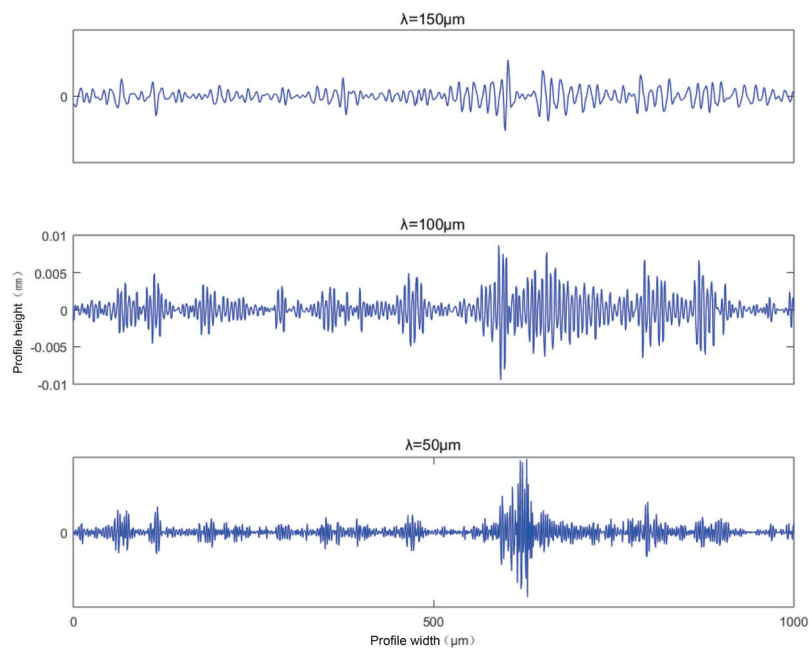

Fig. 1. (Color online) Multiscale registration of trace signals.

2.3 Curve profile mapping to the embedded layer

The parametric curve outline is set to β (β: $D \rightarrow \mathbb{R}^n$), where D is the parameterized determination domain and $\mathbb{R}^n$ is a real number set, defining $\|\cdot\|$ as the Euclidean 2 norm in $\mathbb{R}^n$. Then, a continuous mapping F: $\mathbb{R}^n \rightarrow \mathbb{R}^n$ is defined, and the shape q: $D \rightarrow \mathbb{R}^n$ of the β is defined using the square root velocity function, where

$$q(t) = F\left(\dot{\beta}(t)\right) = \frac{\dot{\beta}(t)}{\sqrt{\left\|\dot{\beta}(t)\right\|}}. \tag{1}$$

For each $q \in L^2(D, \mathbb{R}^n)$, there will always be a β that can be defined by the square root velocity function of q obtained by $\beta(t) = \int_0^t q(s)\|q(s)\|ds$. Then, the curve can be scaled to unit length by $\int_D \|q(t)\|^2 dt = \int_D \|\dot{\beta}\|^2 dt = 1$ to achieve scale invariance, which is expressed as the unit hypersphere point in the preshape space $L^2(D, \mathbb{R}^n)$. Finally, by defining the distance between two curves by minimizing the length of the geodesic between the points in the preshape space, the geodesic is calculated by the analytical expression and the path correction algorithm. The preshape space rotation and re-parameterization invariance are implemented through singular value decomposition and dynamic programming, respectively.

3. Trace Similarity Matching Deep Learning Model

3.1 CNN model

Since the input sample is a profile (or segment), the convolution and pooling layers are one-dimensional signal inputs. Batch normalization is carried out after each convolutional layer to reduce dependence on the input normalization and initialization of the network. We evaluate and set the convolution size, the number of maps, and the pooling layer size. At the same time, the average pool and ReLU activation function are introduced to speed up the training and reduce the impact of the gradient disappearance. The optimization is performed using stochastic gradient descent, and finally the trained trace feature CNN model is used for similarity recognition.

Combining a test toolmark with a sample toolmark in the sample pool as a task, each test sample was randomly placed in a thread pool. The number of thread pools and the number of concurrents take full advantage of the current CPU core number. In each task, a different matching strategy is performed; the calculations in all tasks will be merged in subsequent steps (Table 1).

3.2 Triplet loss and similarity calculation

We randomly select a sample (x_{p1}) from the training dataset (space) and then a sample (x_{p2}) belonging to the same class as x_{p1} and a sample (x_n) from a different class, thereby forming

Table 1
Architecture of CNN branches.

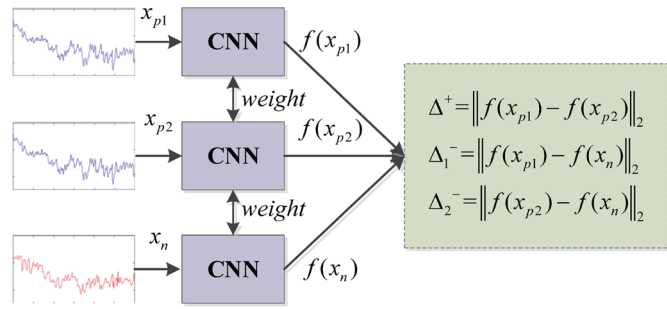| Layer #	Description
1	spatial convolution $(1,5) \rightarrow 64$
2	spatial batch normalization
3	ReLU activation function
4	average pooling $(1,3)$
5	spatial convolution $(1,5) \rightarrow 32$
6	spatial batch normalization
7	ReLU activation function
8	average pooling $(1,3)$
9	dropout
10	linear $\rightarrow$ n feat
11	Tanh activation function

Fig. 2.   (Color online) Trace similarity matching deep learning model.

a triplet $T = \{x_{p1}, x_{p2}, x_{pn}\}$. $f(x_i)$ is the normalized height of the embedded features, while the dimensions of the embedded layer are controlled by the size of the final layer of network branches.

The distance between all samples is utilized and the Softmax layer and root mean square standard are used to achieve the condition that Δ^+ is not only less than Δ_1^+ but also less than $\Delta^* = \min(\Delta_1^-, \Delta_2^-)$ to simplify the training sample selection process. The loss is defined as

$$\ell(T) = \left(\frac{e^{\Delta^+}}{e^{\Delta^+} + e^{\Delta^*}} \right)^2 + \left(1 - \frac{e^{\Delta^*}}{e^{\Delta^+} + e^{\Delta^*}} \right)^2 \tag{2}$$

The L^2 norm is used to estimate the distance between the traces in the embedded layer, and the loss function is used to minimize the local difference between matching traces, thereby completing the similarity calculation (Fig. 2).

3.3   Triplet selection and data enhancement

Since the weights of all three branches in CNN are shared, all feature extractors in these three samples are identical; that is, equal sampling results in an equal representation in the embedding. This may lead to several effects: firstly, if local features are exhibited in all three samples, this feature does not improve the loss and is therefore suppressed by CNN. Secondly, if local features appear only in positive samples rather than in negative samples, CNN will use this local feature to separate samples.

Negative and positive samples are randomly arranged using the same factor. Since the location of local features is a decisive factor for tool identification, two new artifacts are created during the training period using matching and mismatched tool trace profiles. It is essential that rankings occur simultaneously throughout the triplet. If a separate negative sample is arranged, the position of the stitching artifact, i.e, the seam, will be learned by CNN to distinguish between positive and negative samples. If all three samples are independently

arranged, the exact location of local features on the profile is suppressed and cannot be used for tool trace differentiation. The number of possible triplets is increased by moving the individual features to different locations, thereby indicating that CNN not only detects local features, but also observes their position on the profile.

Instead of training the whole profile, segments are randomly cropped from input signals. Since this introduces no seams, the positive and negative samples can independently be distinguished. Owing to a smaller input, however, the overall number of CNN parameters decreases. To compute the similarity value of two profiles during evaluation, a sliding window approach is applied to profiles cropped from the center of a signal.

3.4   Implementation details

Similarity scores are computed by calculating the L^2 distance between the representations of the traces in the embedding. For the profile segment, a sliding window is used to compute multiple representations, that is, one for each segment or patch, from top to bottom. The sum of the pairwise distances of the corresponding representations is then used as the distance measure between two traces. The step size is set to 1/8th of the height of the segment or patch.

The optimization parameters are set through multiple experiments: the optimization is performed using stochastic gradient descent with a learning rate of 0.0001, a weight decay of 10^{-4}, and a momentum of 0.89.

4.   Experimental Testing

4.1   Experimental setup

The automatic classification experiment of the actual line cutting tool is carried out to verify the performance of the proposed method. The experimental environment is set up as follows.

For the real simulation of crime scene data collection, 100 common cutting tools such as pliers, bolt cutters (Fig. 3), and dampers were selected and 1-cm-diameter copper bars were cut at five angles of 15, 30, 45, 60, and 75° to form 500 breakages (Fig. 4). All traces of a particular

Fig. 3.   (Color online) Cutting tools.

Fig. 4.   (Color online) Breakages to be analyzed.

α are put in the test set*;* all others put out in the training set. The naming of partitions reflects the characteristics of traces in the test set; for example, T15 contains all traces with $\alpha = 15°$ in the test set. These partitions enable the evaluation of the performance of the proposed method for finding matching tools in the annotated and trained database.

The following are the sampling parameter settings used: laser detection resolution, 1 μm; minimum step distance, 1.25 μm; subdivision number, 3200 steps/s; number of same-frequency pulses, 1000; sampling interval, 50 ms; sampling frequency, 20 Hz; stage repeat positioning accuracy, 0.005 mm. Five tests were performed on each of the broken ends, and a total of 2500 sets of test data were formed. The matching algorithm is written in Python and runs on a deep learning host with an Intel i7 7700 k CPU, an nVidia GTX 1080 Ti GPU, and 8 GB DDR4 RAM.

4.2   Performance evaluation

Considering the scenario of an actual case, it is more efficient to only search through n feasible signals than look through thousands of detected trace signals. The top-n soft criterion is defined as

$$\text{top-}n = \frac{\sum_{query} M}{N},$$

(3)

where M represents the matches in the top-n results and N represents the number of queries. This score may have the following two disadvantages: 1) It cannot tell how many of the relevant traces are found, but can only tell whether any have been found. 2) The performance must be evaluated with multiple top-n scores. To overcome these disadvantages, the mean average precision (MAP) is proposed, which is defined as

$$MAP(Q) = \frac{1}{|Q|} \cdot \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{i=1}^{m_j} precision(R_{ji}),$$

(4)

where information requires $q_j \in Q$, the set of all information requires Q, relevant documents $\{d_1, ..., d_{mj}\}$, and R_{ji} is the minimal set of ranked retrieval results containing d_i. In the case of tool trace detection signals, q_i could be formulated as "finding the traces formed by the same tool as the supplied trace," and d_i as "traces formed by the same tool." The retrieval results for each q_i are sorted by similarity score. Each R_{ji} then contains both d_i and all other traces, which are more similar to the supplied traces than d_i. A full score of 1.0 is achieved when all d_i are at the top; therefore, all R_{ji} only contain relevant documents.

4.3   Results and discussion

As shown in Fig. 5, the blue and red waveforms are repetitive detection signals of the same tool trace. The case of the smallest difference between the two datasets is shown in the third column. The overlay of the two overlapped signals is shown in the fourth column when the match is completed. Visual observation shows that the signals have a high degree of overlap.

In Table 2, our baseline is compared with the results published by Baiker *et al.*[6] As can be seen in this table, for the method from Ref. 6 using the 45 datasets, which similarly contains only comparisons with α differences of 15 and 30°, a MAP of 0.69 is achieved. With increasing α differences, the MAP drops to 0.45 for T15 and even to 0.31 for T75, which both contain α differences of 15 to 60°. This indicates that the proposed method is more suited for processing α differences greater than 15° than Baiker *et al.*'s method. For T45 and T30, the MAP exceeds 0.9, which indicates that most of the matching tools are ranked at the top. For T15 and T60, the MAP drops slightly to 0.77 and 0.83. However, for T75, even if the α difference distribution is the same as that for T15, the MAP only reaches 0.52. This can be explained by the degradation of the traces at greater α values.
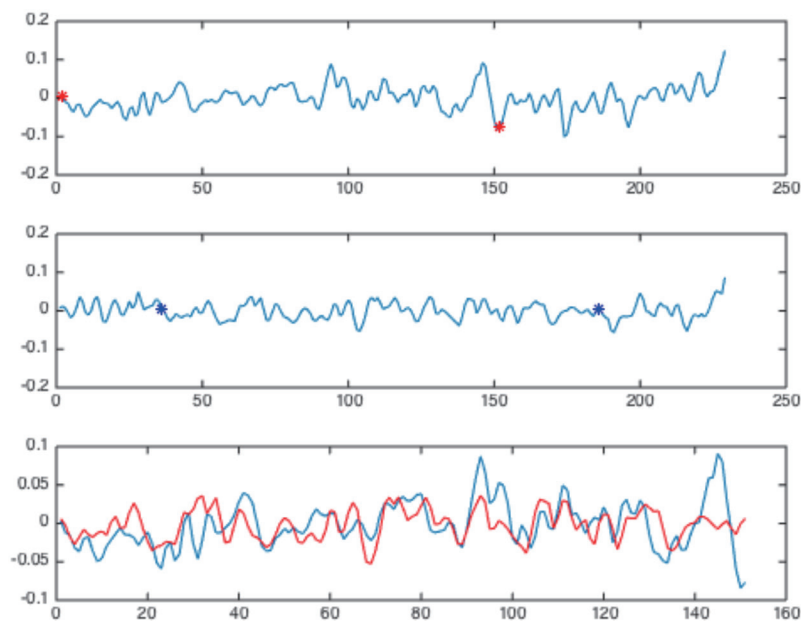

Fig. 5.   (Color online) Matching of two trace detection signals.

Table 2
Results for proposed method and comparison with Ref. 6

Dataset	Metric	Method from Ref. 6	Proposed method
T15	MAP	0.45	0.77
T30	MAP	0.68	0.95
T45	MAP	0.69	0.93
T60	MAP	0.53	0.83
T75	MAP	0.31	0.52

5.   Conclusions

In this study, we focused on shearing section trace matching based on elastic shape metrics and deep learning. Actual trace inference cable cutting tools were analyzed. The isolated forest algorithm was applied to perform abnormal data correction. Then, a multiscale registration framework was used to extract the outline of the trace coarse-transition curve, and the square-speed function was introduced to optimize the elastic shape measurement algorithm to complete the curve profile embedding layer mapping. A trace feature parameter sharing joint triple deep learning model is developed, with optimal triplet selection and data enhancement strategies, using the L^2 distance for embedded layer similarity calculation.

Even if the dataset is small, the performance of the proposed method is good. The network can adapt to α differences of 15–60°, and the MAP of the T15 partition is 0.77. For α differences of 15–45° in the T30 partition, a MAP of 0.95 is achieved. Nevertheless, this method requires further improvement for the T75 dataset.

For future work, the performance could be improved by replacing the sliding window method for distance calculations. In addition, increasing the size of a dataset will allow a more detailed evaluation.
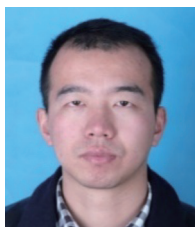
Acknowledgments

References

1   S. Bunch and G. Wevers: Sci. Justice **53** (2013) 223.
2   N. Michael: IEEE Trans. Image Process. **3** (2006) 624.
3   S. Chumbley, S. Zhang, M. D. Morris R. Spotts, and C. Macziewski: Forensic Sci. **62** (2017) 83.
4   A. B. Lock and M. D. Morris: Technometrics **55** (2013) 548.
5   C. Gambino, P. Mclaughlin, L. Kuo, F. Kammerma, P. Shenkin, P. Diaczu, N. Petraco, and J. Hamby: Scanning **33** (2011) 272.

6   M. Baiker, I. Keereweer, R. Pieterman, E. Vermeij, J. V. D. Weerd, and P. Zoon: Forensic Sci. Int. **242** (2014) 186.
7   N. Pan, X. Wu, and Y. Liu: Chin. J. Sci. Instrum. **36** (2015) 1372.
8   R. Spotts, L. S.Chumbley, L. Ekstrand, S. Zhang, and J. Kreiser: Forensic Sci. **60** (2015) 303.

About the Authors

Nan Pan (from Huaiyuan, Anhui Province; Ph.D., associate professor, and master instructor) received his B.S. degree in 2008 from Anhui University of Science & Technology and his Ph.D. degree in 2012 from Kunming University of Science & Technology. He is now an associate professor and master instructor in Faculty of Civil Aviation and Aeronautical, Kunming University of Science & Technology. His main research interests include signal processing and criminal investigation special equipment R&D. (15808867407@163.com)

Yi Liu (from Leshan, Sichuan Province; senior engineer), graduated from Baicheng Normal University in 1998. He is now the general manager of Kunming SNLab Technology Co., Ltd. His main research interests include forensic science equipment R&D. (yiliu@camsonar.com)

Dilin Pan (from Huaiyuan, Anhui Province; professor and master instructor) received both his B.S. degree in 1982 and his M.S. degree in 1986 from Anhui University of Science & Technology. He is now the chief engineer of Kunming SNLab Technology Co., Ltd. His main research interests include signal processing and forensic science equipment R&D. (dilinpan@camsonar.com)

Xin Shen (from Qujing, Yunnan Province; professor-level senior engineer) received his B.S. degree in 2003 and his M.S. degree in 2012 from Kunming University of Science & Technology. He is now an engineer of Metrology Center of Yunnan Power Grid Co., Ltd. His main research interests include signal processing and internet of things. (23755803@qq.com)

Xin Wang (from Kunming, Yunnan Province; Senior engineer) received her B.S. degree in 1988 from Kunming University of Science & Technology. She is now an engineer of Metrology Center of Yunnan Power Grid Co., Ltd. Her main research interests include signal processing and instrumentation. (wangxin@qq.com)