Delft University of Technology

Reducing manual labour in forensic microtrace recognition with deep learning

Rijpkema, Gerben; Kalisvaart, Dylan; Korovin, Serafim; Spengler, Daniel; Pals, Anna; van der Weerd, Jaap; Smith, Carlas S.

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Reducing manual labour in forensic microtrace recognition with deep learning

Gerben Rijpkema [a], Dylan Kalisvaart [a], Serafim Korovin [a], Daniel Spengler [a], Anna Pals [b], Jaap van der Weerd [b], Carlas S. Smith [a],*

[a] *Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands*
[b] *Netherlands Forensic Institute (NFI), The Hague, The Netherlands*

ARTICLE INFO

ABSTRACT

Forensic microtrace investigation relies on time- and labour-intensive microscopic analyses. To aid forensic experts in their investigations, an image recognition model for microtrace localisation and classification is needed. In this work, we use deep learning to automate trace recognition in images captured with automated microscopy. We localise and classify fibres, hairs, skin, glass and sand in microscopy scans through pixel-wise classification of tape-lift samples. As deep learning requires extensive amounts of annotated training data, we additionally investigate various pretraining strategies to minimise the required annotation workload. We compare ImageNet pretraining, pretraining with self-supervised learning and a sequential application of these approaches. We find that pretrained models are able to reduce the required annotated data twofold compared to models trained from scratch while retaining the prediction accuracy. While our ImageNet-pretrained models outperform our self-supervised-pretrained models, we achieve the highest accuracy by combining the two approaches, resulting in a factor 4 reduction of manual annotated microtraces or a 65 % improvement in recognition and localisation accuracy (mean intersection over union increases from 0.34 to 0.56 due to pretraining) when training on only 2.2 dm^2 of annotated tape lift scans. The developed models offer a solid fundament for automated analysis of forensic microtrace scans.

1. Introduction

Microtraces such as hairs, skin cells and fibres often aid in the reconstruction of a crime by providing information on items, locations, people and their actions [1]. Microtraces need to be recovered from their carrier to allow a detailed analysis. One of the methods used to recover these traces from areas of interest is tape lifting. Tape lifting is a fast technique and allows the recovery of different types of traces [2]. A specialised method to lift traces from an item or body is one-to-one taping [3,4]. In this method, the item or body is fully covered with transparent tapes, such that the original location of the traces identified on the tapes can be retrieved. This may provide important clues in activity level interpretations [5].

Tape lifting is a quick and efficient method to recover traces. In the next step, the recovered traces need to be examined. The current routine approach for further examination is based on microscopy [2,6, 7]. In this approach, a microscopist scans the tapes and assesses each trace individually. The cited papers indicate that this approach can lead to very powerful results. Examination of traces on tapes is generally expensive, as it is a time-consuming task carried out by highly trained examiners and large areas of tape have to be investigated. According to De Wael et al. [4], one-to-one taping of a body on average leads to about 1 m^2 of tape lifts. Such tapes may contain several thousands of traces. This leads to long analysis times.

Early systems, called 'fibre finder systems', were introduced a few decades ago [8], to reduce analysis time. However, these systems have not become part of routine practice in forensic laboratories. A more recent step in automation has been made with the procurement of automated microscopy systems within the *Shuttle* project [9] to capture digital scans of tape-lifts. The scope of the instrumentation developed within this project has been broadened when compared to the earlier 'fibre finders': it is based on a combination of microscopic techniques (including reflection, transmission and polarisation modalities) and aims to visualise several types of microtraces (fibres, glass, blood, skin, sand). Due to this broad approach, automation is expected to reduce manual labour and cost and hence improve the effectiveness of trace evidence investigations.

Current digital microscopes, including the microscopes developed in *Shuttle*, acquire images. A current challenge is recognising and localising microtraces in the captured scans automatically and rapidly.

A machine learning approach is promising for this application, considering the various types of microtraces that can be expected on tape lifts, and their representation as images. Analysis of images by a machine learning approach is nowadays standard and powerful [10], but applications to forensic microtraces are limited to the classification of gunshot residues and tapes [11,12]. The authors did not find models that are able to identify several types of microtraces from their microscopic images.

Therefore, we develop methodology for segmentation and classification in microscopic scans and show its ability to precisely localise and classify traces on tape-lifted microtrace samples. This model is based on annotations provided by experts and creates overviews that show the identity and distribution of some of the most useful microtraces. These overviews assist experts in deciding on further investigation procedures.

The developed methodology is based on a deep learning approach. Deep learning image recognition relies on training a neural network to derive meaningful information from input images. Neural networks learn complex patterns and relationships within images. As a result, deep learning has achieved remarkable success in visual tasks such as image classification and semantic segmentation [13–15]. Typically, the network is trained with a large database of annotated images [14–18].

Our developed methodology includes a deep neural network for automated trace classification, schematically represented in Fig. 1. In Fig. 1a, we classify pixels in microscopy scans as either fibre, hair, skin, glass, sand or background. The network is composed of a ResNet-50 [14] feature extractor that embeds the relevant information of the input image into a feature representation. This feature representation is subsequently used by a fully convolutional [19] pixel classifier to assign a prediction for each pixel (semantic segmentation). The network is trained from scratch by initialising the weights with random parameters and training only via annotated microtrace images. Alternatively, we use an ImageNet-pretrained feature extractor (see Fig. 1b) or pretrain the feature extractor with SSL on unannotated microtrace images (see Fig. 1c). In Fig. 1d, we show our method for extracting and augmenting training images from microtrace scans.

Generally, the quality of a deep learning model improves with a larger set of annotated images. As an example, ImageNet, a well-known database, contains over 1 million annotated images that can be used to train image recognition models. These images show everyday objects and cannot be used to directly train a system to recognise images of microtraces. As a result, the development of a trace recognition model requires the creation of a database of expert annotated microtrace images.

As an additional requirement, forensic experts are interested not only in the presence or absence of specific traces but also in their location or spatial distribution. Hence, the annotations should also provide the segmentation and location of the traces. This implies that each pixel in the training data has to be annotated. Altogether, a large number of tapes has to be annotated by experts, as a variety of traces can be encountered with tape lifts and the number of traces per sample is low [20]. This makes the compilation of an annotated database a labour-intensive process that needs to be carried out by experts who can recognise and localise the relevant traces. The cost to compile a database may thus become extremely high [15].

Therefore, we additionally investigate methods to minimise the expert annotation workload through pretraining. In a pretraining task, the neural network is trained either on relevant images from other sources, or on unannotated images. It is anticipated that pretraining reaches the same trace recognition capacity while using fewer annotated images.

The first pretraining method used in the current study is based on ImageNet classification [21,22]. As stated before, ImageNet cannot be used to train a network to recognise microtraces. Nevertheless, training using ImageNet teaches the network to recognise visual features important for human observers, such as the colour, shape, and texture of objects in the input images. As the network already recognises such important features, recognition of additional items, such as microtraces, may become easier. Indeed, it has been shown that a pretrained neural network outperforms a randomly initialised network with regard to convergence speed and accuracy [23]. Furthermore, ImageNet pretrained models are widely available, and no computational efforts are needed to use a pretraining neural network [24]. However, the benefits of ImageNet pretraining diminish for tasks dissimilar to the classification of everyday photographs, as the learned visual features can be sub-optimal for other domains [22,25].

The second pretraining method is based on images of microtraces that have not been annotated. This approach is based on self-supervised learning (SSL) and uses the framework proposed in [26]. In this approach, two different and perturbed versions are made for each unannotated microtrace which are called *augmentations*. The two augmentations are made via cropping, recolouring and rotating. Pretraining aims to minimise the distance between the features of the two augmentations. In this way, the neural network learns what features of an image are important for microtrace classification and are to be used when microtraces are observed. In SSL, the system is trained on augmented images, without knowing the nature of the displayed images. Therefore, pretraining by SSL does not require the annotation of images.

Finally, we combined the pretraining approaches: a neural network is pretrained using ImageNet pretraining. Subsequently, the network is further pretrained using SSL, to optimise the processing of microtraces. This approach follows recent research on pretraining in SSL [27].

2. Methods

2.1. Samples and datasets

Textile, glass and sand samples were taken from the general collection available in the authors' laboratory. Dandruff, donated by volunteers was used as skin cells. The used tapes and tape backings were provided by Spectricon (Chania, Greece). Samples were prepared by distributing materials on a solid surface and lifted using the tape. Afterwards, the tape was attached to the provided backing to prevent contamination. Images of the tapes were acquired using a SMMART automated microscope, developed by Spectricon, Chania, Greece [28], within the context of the *Shuttle* tender [9]. Samples were imaged using transmitted light and acquired as colour images, using the calibration and acquisition procedures proposed by the manufacturer. Each pixel in the resulting images represents an area of approximately 1×1 μm. Images were corrected for shading effects and stitched together using routines written in Python. In this way, a tape of 80×80 mm results in a `.tiff` file that contains 6.4 Gigapixels. In total, 324 tape samples of various sizes were scanned. Ten of these were annotated (see Table 2).

2.2. Annotation

Traces present in a selection of the acquired images were segmented by thresholding and preliminarily classified using simple heuristics, such as the colour histograms and selected shape features. The preliminary annotations were saved as geojson objects and opened in QuPath software [29] together with the original images, where experts manually corrected the preliminary annotations. In this way, the workload for the experts was minimised. The time spent on correcting and manually annotating trace images is estimated around 120 h. In this period, a dataset of approximately 21.000 annotated image patches (see Table 2) was created. Table 3 presents the occurrence of each trace annotation in the training dataset.

The annotated dataset was split into training and test sets. The test set consisted of an 80×80 mm tape lift containing all trace types. Nine other annotated tapes were used as training set. In total, the tape lift area of the training set was approximately three times as large as the test set, as shown in Table 2.
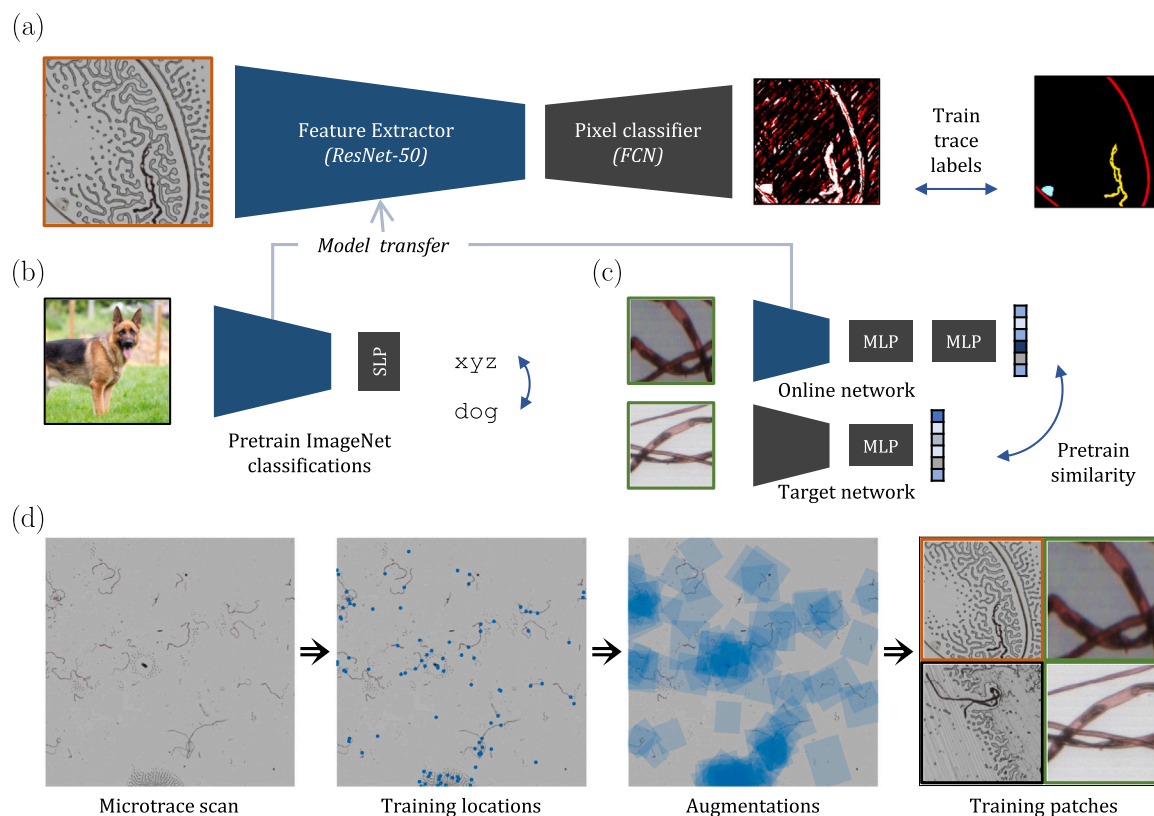
Fig. 1. Schematic overview of deep neural network for automated microtrace classification. (a) Training trace predictions with expert annotations. A residual network with 50 layers (ResNet-50) [14] feature extractor is used in a fully convolutional network (FCN) [19] structure for pixel-wise classification. The network aims to find the relationship between the microtrace images (left) and the corresponding annotation provided by experts (right) by minimising the cross-entropy loss between the predictions and the annotations. The microtrace image in the figure shows a hair, a fibre and air bubbles. (b) ImageNet pretraining of feature extractor through classification of annotated everyday photographs in the ImageNet database [21] with a single layer perceptron (SLP) classifier. (c) Pretraining the feature extractor with the self-supervised learning (SSL) model BYOL [26] on pairs of unannotated images. Here, the weights of the feature extractor are adapted such that pairs of similar images yield similar features. The architecture is composed of two ResNet-50 networks and a set of multi-layer perceptrons (MLPs). (d) Extraction and augmentation of training images from tape-lift scans. For pretraining with SSL, image pairs are extracted. For training trace predictions, image patches are extracted together with their corresponding annotation mask.

2.3. Training

A schematic representation of the neural network used for microtrace classification is provided in Fig. 1a. A detailed overview of the architecture is provided in Supplementary Figure S7. This is based on the procedure proposed elsewhere [26,30]. The network consists of a fully convolutional network (FCN) architecture [19] with a ResNet-50 feature extractor [14]. The last block of the feature extractor (*stage 5*) uses dilated convolutions to increase the resolution of the feature map, which may benefit semantic segmentation [31].

The network is trained by minimising the difference between the predicted pixel classifications and the expert annotated classifications via standard per-pixel softmax cross-entropy loss [24,32]. We do not freeze the weights of pretrained feature extractors but instead allow all network weights to be optimised.

In benchmark tests, the feature extractor was randomly initialised. Supplementary Table S1, S2 and S3 respectively provide the hyperparameters, augmentations and initialisations used in training.

2.4. ImageNet pretraining

For the ImageNet pretraining approach, the neural network architecture and training procedure are identical to the procedure presented in Section 2.3. However, the ResNet-50 weights are not randomly initialised. Rather, they were initialised by the model provided by [24]. This procedure is visualised in Figs. 1a and 1b. Fig. 1b shows the training of a model based on the 1.3 million images of ImageNet [21]. This training was carried out by [24]. The trained network was downloaded and its weights were used to initialise the network for training the classifier using microtrace images. This transfer is illustrated by the arrow between the subfigures of Fig. 1. The hyperparameters used by [24] to obtain these weights are summarised in Supplementary Table S1.

2.5. Self-supervised learning

Fig. 1c shows pretraining of the feature extractor using SSL on image patches extracted from unannotated microtrace scans. We employ the SSL model "Bootstrap Your Own Latent" (BYOL) [26]. With BYOL, the feature extractor is rewarded for predicting the similarity of the feature representations of two images with the same underlying structure.

These pairs of images are obtained by augmentation, i.e. alteration of an image of a single trace. For this purpose, an image is retrieved using the method described in Section 2.6. This image is augmented twice by using a combination of translation and zoom parameters. Cropping is an essential aspect of creating useful image pairs [26]. Cropping is performed by adjusting translation and zoom parameters, which were consequently optimised for better results. For further details, refer to Supplementary Figure S4.

BYOL is composed of an asymmetric architecture of two separate networks: an online network and a target network, as shown in Fig. 1b. The online network is composed of a ResNet-50 feature extractor,

an multilayer perceptron (MLP) projector and an MLP predictor. The target network is composed of a second feature extractor and a second projector, both with different weights than the online network. Supplementary Figure S8 provides a detailed overview of the implementation and the output dimensions of each network.

During SSL, the online network and target network each take one of the augmented patches as input. The weights of the online network are optimised to predict the output of the target network.

In this way, the model learns to recognise the similarities between two cropped parts of a single trace image and, hence, to recognise the important features on which the recognition can be based. SSL does not require a classification annotation of the trace. As a result, it can learn from images that have not been annotated. The implementation of SSL is discussed in detail in Supplementary Note 1.

2.6. Image extraction

The training (see Section 2.3) and SSL (see Section 2.5) make use of image patches. These image patches are extracted from transmission microscopy scans as shown in Fig. 1d.

Tape lifts generally contain a variety of small traces scattered on a transparent tape. The dimensions of the used tapes are 80×80 mm, which is very large compared to the size of common microtraces and to the optical resolution of the used microscopy system. This results in large datasets containing around 6.4 Mpixel. The tapes are usually not completely covered by microtraces, but also contain background areas. In fact, experts annotated less than 2% of all pixels as trace (see Supplementary Table S4). The large size of these scans prevents calculations on a full scan, as the required computing power would be immense. Instead, image patches are sampled from the scans.

Random sampling image patches from the microtrace samples [15] would result in many patches containing only background, which is not useful for training.

Fortunately, most of the background area can easily be recognised. The used tape has an excellent light transmission and the background is shown as homogeneous white areas. Traces generally scatter and absorb light and are thus displayed as dark objects. Therefore, thresholding with a single global threshold suffices to distinguish foreground areas from background [33]. The threshold value is determined using the histogram-based triangle method [34], which is suited for images dominantly consisting of background [33,35,36]. This is shown in Supplementary Figure S11. With the found threshold, a total of 94% of the image area in the annotated dataset is estimated as background. The remaining 6% is estimated as foreground, including possible microtraces or artefacts. The proposed procedure extracts patches centred around foreground areas. As a result, filtering out background-rich areas later in the process, such as proposed by [23,37] is not necessary. Training coordinates are then sampled uniformly at random from the thresholded foreground areas. At each sampled coordinate, an image patch of size 256×256 pixels is extracted in a resolution of 4 μm per pixel.

For trace training, single image patches are extracted together with corresponding annotation masks. These image patches are augmented to artificially enlarge the dataset. Specifically, we make a crop of random zoom, aspect ratio and rotation and resize it to 256×256 pixels as shown in Supplementary Figure S12. Then, we further augment the image patches by recolouring, blurring and mirroring, which results in the required training images. The chosen augmentations, randomisation processes and their parameters are derived from [26] and are listed in Supplementary Table S2.

The chosen FOV of 1024×1024 μm per patch ensures that the majority of the traces is captured fully within the FOV (see Table 3). Images are mean-downsampled to a resolution of 4 μm/pixel, resulting in a patch size of 256×256 pixels.

2.7. Evaluation

Trained models are tested by evaluating their predictions on an unseen microtrace scan. The used scan was annotated, but was not part of the annotated training dataset or the unannotated dataset used in SSL pretraining. This scan consists of an 80×80 mm tape lift sample containing all trace types, shown in Supplementary Figure S14. During testing, the scan is processed via a uniform grid of image patches (see Supplementary Figure S5), without oversampling the foreground with thresholding or applying augmentations as described in Section 2.6. During testing, the field of view (FOV) of each image patch is enlarged by 12.5%, creating partial overlap between neighbouring patches and ensuring full coverage of the scanned area. Predictions at the outer edges of each patch are then discarded, as the network needs surrounding image context to correctly interpret features near the borders; without this context, edge predictions are less reliable [32,38].

The trace recognition and localisation performance of the model is evaluated with the Intersection over Union (IoU) [15,39]. This approach provides a powerful way to assess robustness, and combines various parameters, such as recall and specificity. For any class A, the IoU is defined as the number of correctly identified trace pixels divided by the sum of all pixels either predicted or annotated as class A:

$$\text{IoU}_A = \frac{\text{TP}_A}{\text{TP}_A + \text{FP}_A + \text{FN}_A}, \tag{1}$$

Here, TP_A denotes the number of true positives, while FP_A and FN_A denote the number of false positives and false negatives, respectively. The maximum value of $\text{IoU}_A = 1$ indicates that the annotations and predictions match perfectly. Conversely, the minimum score of $\text{IoU}_A = 0$ indicates no true positives were predicted.

To quantify the overall performance over all classes, the mean of the IoU (mIoU) of the n_c non-background classes is taken:

$$\text{mIoU} = \frac{1}{n_c} \sum_{A=1}^{n_c} \text{IoU}_A \tag{2}$$

In our case, $n_c = 5$, with the classes being fibre, glass, hair, sand, and skin. Similar to the evaluation protocol of [16], we calculate the overall mean Intersection over Union (mIoU) based on the accumulated statistics over the entire test set. As single image patches typically do not contain all of the five trace classes, the accumulated statistics are a more generalisable representation of the trace recognition performance. The violin plots used throughout the report show the distribution of mIoU values that are obtained by accumulating over a randomly determined 10-fold split in the test set that was kept constant throughout the experiments.

mIoU is calculated by averaging over the classes present in the test sets. This means that the IoU of each class contributes equally to the mean regardless of how many examples of each class are present in the test dataset. This approach is also known as micro-averaging [40], in contrast to macro-averaging, which aggregates the IoU scores across all instances in the test set. Macro-averaged IoU can be more lenient towards biased classifiers in imbalanced datasets due to its aggregation of class-wise IoU values. Therefore, it is advised to employ a micro-averaging approach for unbalanced datasets. Moreover, we display violin plots with the mIoU distributions and show the confusion matrices (Supplementary Figure S1) to provide additional insight into the model performance.

To investigate annotation-efficient learning, we split our annotated training set into equally sized regions as shown in Supplementary Figure S15 and vary the number of regions used in training the trace classifications.
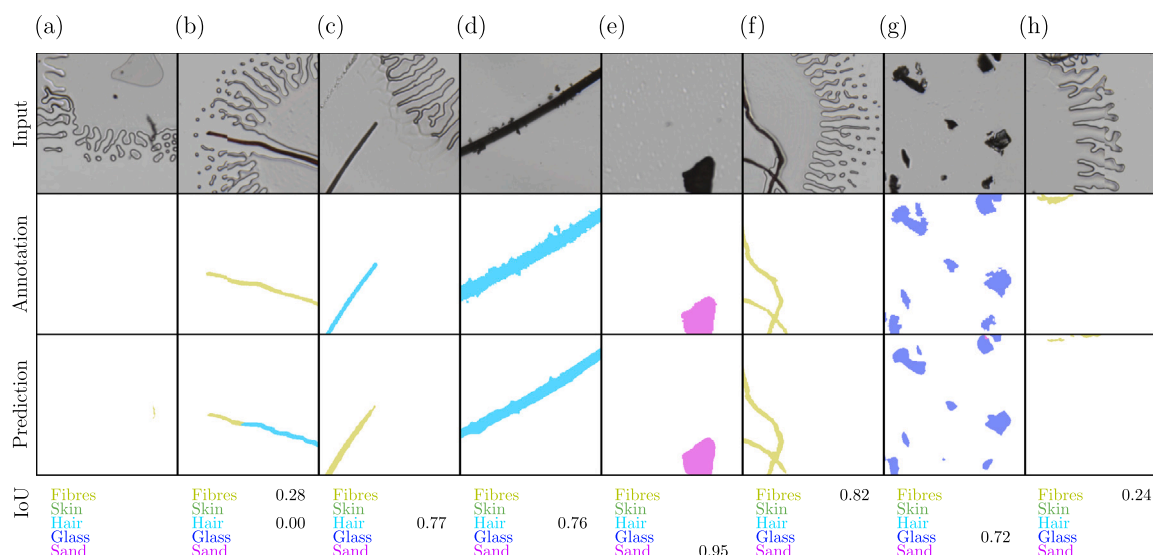
Fig. 2. Comparison of model predictions with expert annotations for patches in the test set to visually assess model performance and corresponding IoU values. The IoU values for classes that span less than 1% of the total image area in both the expert annotations and the predictions are omitted.

3. Results

Four models were trained, following the general procedure in Fig. 1, namely models with random initialisation, with ImageNet pretraining, with SSL pretraining, and with the combination of both pretraining methods. Our results show that the final option, combining both pretraining methods, achieves the best classification, as indicated by the highest mIoU. This training will be highlighted in Section 3.1. Section 3.2 presents our results on the benefit of pretraining, followed by a discussion of our results on SSL in Section 3.3. Finally, we provide insight into the model and its key elements through an ablation study, that is by removing parts of the pipeline. We discuss the results obtained by these ablated methods in Section 3.4.

3.1. Microtrace recognition

The model described in the current section was initialised by an ImageNet pretrained model. It was further pretrained using SSL on roughly 1 m^2 of unannotated tape area. Finally, the model was trained on our annotated dataset of 22 dm^2 of tape area. This training method resulted in an mIoU of 0.56.

A general impression of the meaning of the value of mIoU can be obtained from Fig. 2. This figure provides three rows of images. The top row provides images as acquired by the automated microscope. These images were part of the test set, so they have been annotated by an expert, but were not used during the training of the model. Rather, the trained model was used to identify and localise traces. As a result, the annotations and predictions can be compared. The second row of images provides the annotations. These annotations are colour-coded using the colours shown below the figure. The third row of images provides the predictions generated by the model, using the same colour-coding. Fig. 2a shows an image of air bubbles and other artefacts without annotated traces. Here, the model aligns with the expert in not marking any traces. Fig. 2b shows a fibre. Here, the model classified part of the trace as fibre and incorrectly masked the largest part of the trace as hair. This leads to a fibre mIoU of 0.28 and a hair mIoU of 0.00. Fig. 2c and d show correctly identified hairs. Due to imperfect masks, IoUs of 0.77 and 0.76 are attained respectively. Fig. 2e shows a trace that was classified as sand by both the model, and the expert. The areas indicated by the model and the expert overlap almost perfectly, leading to an IoU of 0.95. Fig. 2g shows an area in which glass particles are annotated and predicted. A visual comparison of these images indicates

that all glass particles were found by both the expert and the model. Nevertheless, the IoU is only 0.72, indicating that the areas shown do not perfectly overlap. In Fig. 2h, the situation is even more extreme. The model predicts the same as the expert annotated. Yet the calculated IoU is only 0.24, as the model missed part of the fibre.

It should be noted that the mIoU is based on the number of correctly classified pixels. Forensic examiners are usually more interested in the number of identified microtraces. As each of the microtraces comprises several pixels in the images, the number of traces found is higher than the mIoU. In practice, we find that nearly all traces are located, though our current methodology does not allow an accurate determination of the number of accurately predicted traces.

Supplementary Figure S2c provides additional image patches on which the model performs poorly. These images were randomly selected from predictions where a class IoU of less than 0.4 was encountered.

Air bubble patterns can be seen in most of the microscopic images. These are caused by air being trapped under the tape. It can be seen from Fig. 2 that the model correctly refrained from identifying these patterns as traces.

Supplementary Figure S1 shows the confusion matrix of the model together with precision and recall analysis. It can be seen that the pixel-wise precision exceeds 60% for each class, indicating that at least 60% of the pixels marked as a certain class are marked as that class by the expert annotator as well. Except for the class Skin, the recall exceeds 60% as well, indicating that at least 60% of the pixels are marked as a certain class by both the expert and the model.

Generating predictions for a microtrace tape of 80×80 mm takes approximately 30 s using our hardware (NVIDIA RTX3090 GPU and Intel i910900X CPU). This is a fraction of the time required for a microscopist to generate an overview of traces on the tape. Also, it is much faster than scanning the tape with automated microscopy, which takes around 40 min with the used device.

3.2. Pretraining for annotation-efficient learning

The current section explores whether pretraining reduces the time-intensive and hence expensive task of annotating images. Our main results are presented in Fig. 3. The horizontal axis in this figure represents the area of tape that was used to train the network. Note that this axis is not linear, as it is based on a sequential doubling of the number of tapes used for annotation. The vertical axis shows the accuracy

(a)

(b)
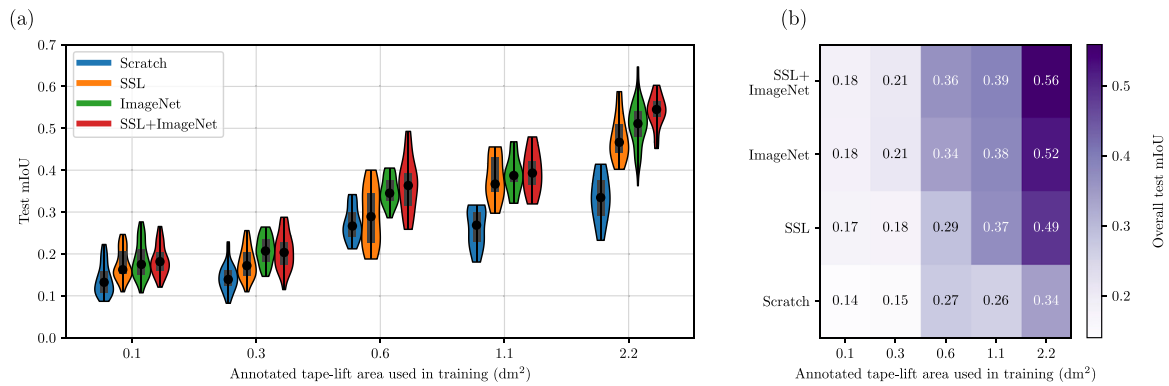
Fig. 3. Benefit of pretraining for annotation-efficient learning. (a) Violin plot showing the distribution of mIoU over a 10-fold split in the test set. The dot presents the median, the inner bar the upper and lower quartile. The line presents the spread between the minimum and maximum. (b) Overall test mIoU per experiment.

Table 1
p-values of label-efficiency results. The shown results are based on training on the full 2.2 dm^2 of annotated tapes. A paired student t-test was used based on the mIoU values in the fixed 10-fold split in the test set.

	SSL+IN	IN	SSL
SSL+IN	–	–	–
IN	0.023	–	–
SSL	0.00025	0.034	–
scratch	$0.37 \cdot 10^{-7}$	$4.3 \cdot 10^{-7}$	$5.4 \cdot 10^{-7}$

of the trained model, as indicated by the mIoU. Training on smaller areas reduces the diversity of the microtraces used to train the model, resulting in less accurate predictions. To reduce the probability of random variation due to the limited sample size, the numbers presented in Fig. 3 for areas of 0.1, 0.3, and 0.6 dm^2 are averaged over 5, 3, and 2 training runs, respectively.

Fig. 3 shows that training on a larger area improves the accuracy of prediction. In addition, the pretrained models achieve higher mIoU values than the models trained from scratch. The ImageNet-pretrained models outperform the SSL-pretrained models. The highest mIoU value is obtained for the model pretrained by both ImageNet pretraining and SSL, followed by training on 2.2 dm^2. In Table 1, it is shown that the difference in mIoU values obtained for training on 2.2 dm^2 are statistically significant.

When the model is trained on 0.1dm^2, pretraining improves the mIoU from 0.14 to 0.18, which is an increase of 29%. The effects of pretraining become even higher when the model is trained on larger areas. When the model is trained on 2.2 dm^2, pretraining improves the mIoU from 0.34 to 0.56 for the combined pretraining methods, i.e. an improvement of 65%. In addition, pretraining followed by training on 0.6 dm^2 of annotated tape or more leads to models that outperform models trained from scratch using two times more annotated data. In fact, combined ImageNet and SSL pretraining followed by training on 0.6 dm^2 of annotated tape leads to a higher mIoU (0.36) than a training from scratch using 2.2 dm^2 (0.34). This means that pretraining, in this specific case, reduced the annotation workload by a factor of four.

Values of mIoU below unity indicate that the classification achieved by the model is not perfect. The types of errors made by the models are not specified in Fig. 3, but can be identified by a further exploration of the data. Supplementary Figure S3 shows a detailed exploration of the various error modes for our experiments. When the model is trained on 2.2 dm^2, the 65% improved mIoU after pretraining can be traced back to a 40% increase in correctly classified trace pixels, a 27% decrease in missed trace pixels and a 56% decrease in false detections compared to training from scratch. The number of confused trace pixels is increased by 1% after pretraining.
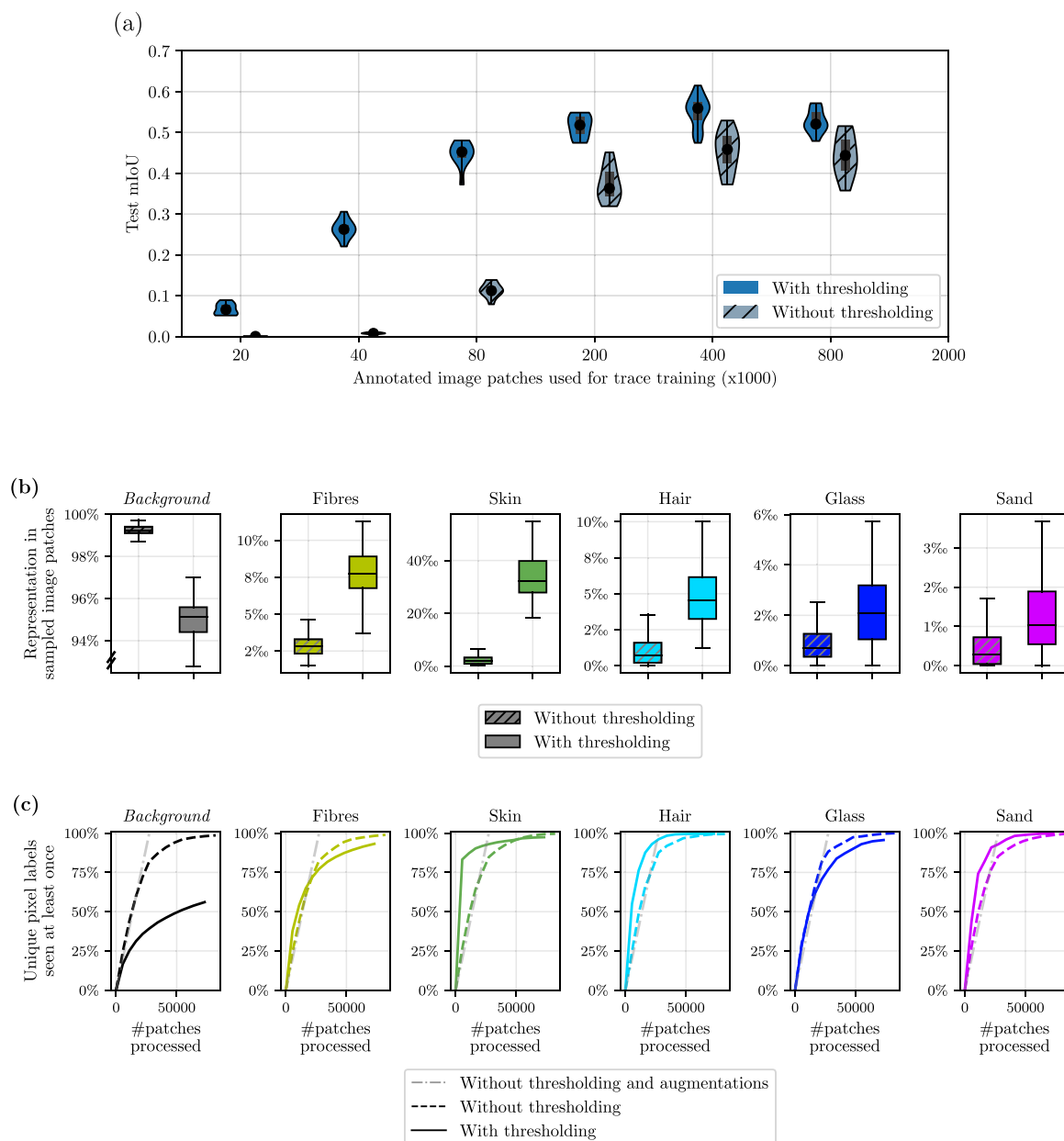
3.3. Self-supervised pretraining

SSL pretraining minimises the distance between the two different images or image augmentations of a similar trace. In the current study, pairs of image augmentations were obtained by selecting a coordinate on a microtrace scan, extracting two images within a maximum distance d_m of this point. Next, each of these images is randomly augmented, according to the procedure described in Section 2.6). Combinations of several augmentation methods have been tested. An overview is presented in Supplementary Table S2.

The results of the SSL experiments of Section 3.2 were obtained with $d_m = 0$ and a maximum magnification augmentation between 0.5× and 2×. These values lead to optimal results, as shown in Supplementary Figure S4a, which shows the obtained mIoU values for different settings of the magnification and translation. An mIoU of 0.42 is achieved with $d_m = 0$ and magnification augmentation between 0.5× and 2×, while choosing a lighter zoom augmentation (between 0.67× and 1.5×), results in an mIoU of only 0.37. It can be anticipated that a light zoom augmentation and no translation results in two approximately similar representations for both views and hence a trivial solution. This reduces the benefit of pretraining.

Optimal parameters for the augmentation are dependent on the class of the trace involved. As stated before, SSL is not aware of the class of the trace involved. Nevertheless, it is possible to pretrain the model using different types of traces. Supplementary Figure S4b-f show the IoU benefit per class to illustrate this effect. This can be illustrated by the results for hairs and glass. A translation of 2048 μm hardly affects the IoU for hair traces (0.61 vs 0.62, see Supplementary Figure S4d), while such a large translation reduces the IoU for glass traces from 0.48 to 0.34 (see Supplementary Figure S4e). This is attributed to the size of the traces. Glass fragments are generally small. Table 3 shows that the glass particles of our dataset generally have sizes of $0.2 - 0.4$ mm. A translation of 2048 μm can therefore cause the trace to move out of the field of view for one of the images in the augmentation pair. This obviously hinders the training, as the SSL procedure assumes that both images display items that should be considered similar. On the other hand, hairs have lengths of more than 5 mm, yielding a larger probability of encountering the trace in both views. The presented parameters are a compromise that offers a balanced performance for all trace types included.

3.4. Influence of image sampling

In the proposed method, training and SSL procedures make use of image patches extracted from transmission microscopy scans. The proposed method to extract the image patches, as described in Section 2.7 involves a thresholding operation, intended to enhance the amount of foreground pixels in the used image patches. Thresholding was considered beneficial due to the large amount of background in

Fig. 4. Benefit of thresholding approach shown in Fig. 1d compared to the uniform approach shown in Supplementary Figure S5. Only trace training is regarded. An ImageNet pretrained-model is used without additional self-supervised pretraining. For the uniform approach, processing 21.000 image patches corresponds to one data set iteration (see Table 2). This is called one *epoch*. (a) mIoU of trace recognition for training with and without thresholding approach. The test mIoU is shown for training with various amounts of image patches across the annotated dataset. It can be seen that the thresholding approach converges earlier and to a higher final accuracy. (b) Efficacy of thresholding approach to oversample the foreground pixels. The class distribution of pixels over 8000 randomly selected image patches is shown. The left boxplots show the distribution of patches sampled without thresholding, the right boxplots show the class distribution for sampling with thresholding. It can be seen that the thresholding approach results in oversampling the non-background classes. (c) Analysis of missed pixels in the annotated dataset due to thresholding. It can be seen that after processing 80.000 image patches (4 epochs for the uniform approach), 95% of the non-background pixels have been seen at least once. Further experiments show that after 400.000 image patches (20 epochs for the uniform approach), 99.7% of all non-background pixels are seen, while 80% of the unique background pixels are seen.

the investigated microscopic scans. In the current section, our method is compared to an alternative method that does not involve thresholding. Rather, this alternative method uses a uniform grid as shown in Supplementary Figure S5.

The results of this comparison are presented in Fig. 4. The horizontal axis in Fig. 4 shows the number of image patches used to train the model. The blue datapoints show the calculated mIoU when using the thresholding method. The performance of the model increases linearly with the number of images used for training, until it levels at an mIoU of around 0.55. If a uniform grid is used, ('without thresholding'), a

lower test mIoU was attained for each of the tested number of training images. After training on 400.000 image patches, the thresholding approach results in a 21% higher overall test mIoU compared to training with a uniform grid. Training on lower amounts of images further increases the performance difference. Specifically, when training with only 80.000 data points, our thresholding approach results in a four-fold increase in mIoU. The improved performance is attributed to a higher representation of traces in the used image patches. Further tests, detailed in Fig. 4b and c, confirm the expectation that thresholding increases the representation of traces.

3.5. Influence of hyperparameters

The current section investigates the hyperparameters for training trace classifications. The sensitivity of segmentation accuracy with respect to hyperparameters is investigated and the chosen values are substantiated. These results were obtained after ImageNet pretraining. Supplementary Figure S6a shows that the mIoU of trace recognition has values between 0.5 and 0.6 for weight decay values between 0 and 10^{-4} but decreases below 0.05 for a decay value of 10^{-2}. Supplementary Figure S6b furthermore shows that mIoU does not significantly change across batch sizes between 5 and 200 when the learning rate is scaled linearly with the batch size.

Images of traces contain many pixels. Each of these pixels is an individual input for our data model. Hence, the time needed to train a model increases with the number of input pixels. Downsampling of the microtrace images may help to limit the calculation times and reduce excess details that can lead to overfitting. On the other hand, downsampling images reduces the details that may be important for classifying traces. Therefore, we evaluated the performance and calculation times of models trained on the original images (1 μm/pixel) and downsampled images (2 and 4 μm/pixel). The results are shown in Supplementary Figure S6c and d. Supplementary Figure S6d shows that downsampling to 2 μm/pixel results in a shorter training time (approximately 4 instead of 15 h) and Supplementary Figure S6d shows an unaffected high mIoU (0.6 mIoU). This results in a speed-up of 3.75 times.

The image patches extracted from the microtrace scans can be altered before they are used to train the model. Such alterations may improve the robustness of the training [26], as reliance on feature diversity in the training data is reduced. Hence, they may improve the quality of the classification [41]. However, excess alteration can cause the simulated feature diversity to transcend the feature diversity of the actual traces, causing a decay in segmentation accuracy. For example, heavily shearing an image of a trace can cause the image to not resemble the trace anymore. Supplementary Figure S6e and f show that altering the magnification and aspect ratio has only a limited effect on the obtained mIoU values. Changing the magnification improves the model (mIoU 0.55 for magnification $0.33\times$ and $3\times$, mIoU 0.52 for a magnification of 1. A limited alteration of the aspect ratio hardly influences the obtained mIoU values. Heavier zoom and aspect ratio augmentation deteriorate the training process. This is probably caused by the induced loss of information by these heavy augmentations.

4. Discussion and conclusion

A data model is presented to classify microtraces shown in microscopic images. The model is based on a residual network with 50 layers (ResNet50). The presented model successfully localises and classifies hairs, fibres, skin, sand and glass traces in the presented, resulting in a mIoU of 0.56. Classification of a data set representing a tape of 80×80 mm requires approximately 30 s.

The model was trained using images annotated by experts. As annotating is a time-consuming task, we explored pretraining methods to minimise the time needed for annotating. A pretrained ImageNet model can be downloaded, so its use does not incur any computational or annotation costs. Pretraining using SSL does not require annotated images, but is demanding computationally. The pretraining approaches result in an mIoU of 0.52 and 0.49 respectively, yielding a +53% and +44% improvement with respect to training from scratch using the same number of annotations. Combining ImageNet pretraining and SSL is more effective than the use of the individual methods and results in the optimal mIoU.

The presented mIoU of 0.56 is based on a model that was pretrained and subsequently trained on annotated data. A model trained from scratch on the same amount of annotated data achieves an mIoU of

Table 2

Overview of datasets. The last column shows the number of unique image patches that can be extracted for a field of view (FOV) of 1024×1024 μm. The unannotated dataset includes the tapes of the annotated training dataset. Thus, a total of 315 samples were scanned.

Dataset	#tapes	Total area	#patches
Unannotated	314	1.082 m²	$1.0 \cdot 10^6$
Annotated			
Training	9	0.022 m²	$21 \cdot 10^3$
Test	1	0.006 m²	$5.8 \cdot 10^3$

Table 3

Traces in annotated training dataset. The class skin is segmented in regions instead of individual cells, therefore yielding the two right columns ill-defined. The trace size is calculated as the median diameter of the minimum bounding circle for each annotation. For the uncertainty measure, the median absolute deviation is used.

Class	Total area	Trace size	Count
Fibres	117 μm²	0.5 ± 0.2 mm	$3 \cdot 10^3$
Hairs	47 μm²	7 ± 4 mm	$2 \cdot 10^1$
Glass	47 μm²	0.2 ± 0.1 mm	$2 \cdot 10^3$
Sand	20 μm²	0.3 ± 0.1 mm	$4 \cdot 10^3$
Skin	79 μm²		

0.34. Our tests indicated that pretraining may reduce the required amount of annotated data by approximately 4.

The finding that ImageNet pretraining outperforms sole SSL pretraining is in line with [37], in which SSL pretraining is investigated for semantic segmentation of biomedical microscopy images, although here another SSL framework (SimCLR [42]) is used. Moreover, it aligns with [43], in which BYOL is compared to ImageNet pretraining for semantic segmentation of natural images [43]. However, [26,44,45], find that BYOL and SimCLR result in higher accuracies than ImageNet pretraining for semantic segmentation of natural images.

SSL is based on pairs of images displaying the same type of trace. In the current study, such pairs were created by extracting different parts from a single image patch of a trace. The image patches have been distorted to improve the robustness of the pretraining [26]. The parameters selected to extract and distort the image patches have been optimised. It is shown that the optimal values depend on the size of the displayed trace. The chosen values are a magnification between $0.5\times$–$2\times$ and a translation of 0 μm.

As tape lift scans typically contain a small number of traces scattered across a large background area, 99% of the image area of our dataset represents background (see Supplementary Table S4). Our proposed image extraction method involving thresholding allows a +0.10 mIoU increase with respect to processing the image area uniformly by effectively oversampling foreground areas.

The presented model exports predictions in the geojson format. The predictions can be displayed along with the original images using QuPath [29] software, which experts have also used to annotate images. This provides a user-friendly method to evaluate the predictions by the model. In this way, the results of the model can be used to initiate further investigations.

The presented model successfully recognises hairs, fibres, skin, glass and sand particles, which are key trace types in forensic work. In this study, all fibres—regardless of colour or material—were grouped into a single "fibre" class, and all crystalline particles into "sand", to maintain a compact classification scheme. While the model uses colour information from the microscopy images, subclass distinctions (e.g., fibre colours or sand mineral types) were not yet addressed. Future work will focus on extending the model to include additional trace types such as blood and pollen, as well as on subclassification within existing categories. These developments may benefit from alternative microscopy modalities, such as reflection or polarisation imaging, to improve discrimination of opaque or complex traces (see Section 3.1).

Our study sets a new baseline for forensic microtrace recognition and gives practitioners insight into the benefits of pretraining, the required annotation workload, augmentation parameters and efficient image extraction for microtrace recognition in tape lift scans. We provide trained models and facilitate analysis of model predictions through a graphical user interface [29] to yield a valuable tool for microtrace recognition that aids forensic experts in their investigation. Hereby, we contribute to automating the microtrace finding process and decreasing the human labour required for trace investigations.

CRediT authorship contribution statement

Gerben Rijpkema: Writing – original draft, Software, Methodology, Investigation, Data curation. **Dylan Kalisvaart:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Investigation, Formal analysis. **Serafim Korovin:** Writing – review & editing, Writing – original draft, Supervision, Software, Methodology, Investigation. **Daniel Spengler:** Writing – review & editing, Writing – original draft, Methodology, Investigation, Formal analysis, Conceptualization. **Anna Pals:** Writing – review & editing, Methodology, Investigation, Data curation. **Jaap van der Weerd:** Writing – review & editing, Writing – original draft, Supervision, Software, Methodology, Investigation, Formal analysis, Data curation. **Carlas S. Smith:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Conceptualization.

Declaration of competing interest

Acknowledgements

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.forsciint.2025.112714.

References

[1] Bernard Robertson, G.A. Vignaux, Charles Berger, Interpreting Evidence: Evaluating Forensic Science in the Courtroom: Second Edition, Interpreting Evidence: Evaluating Forensic Science in the Courtroom: Second Edition, Wiley, 2016, pp. 1–197.

[2] Robertson James, Claude Roux, Kenneth G. Wiggins, Forensic Examination of Fibres, CRC Press, 2021.

[3] T. Coyle, A. Larkin, K. Smith, S. Mayo, A. Chan, N Hunt, Fibre mapping – a case study, Sci. & Justice 44 (3) (2004) 179–186.

[4] K. De Wael, L. Lepot, K. Lunstroot, F. Gason, 10years of 1:1 taping in Belgium — A selection of murder cases involving fibre examination, Sci. & Justice 56 (1) (2016) 18–28.

[5] R. Cook, I.W. Evett, G. Jackson, P.J. Jones, J.A. Lambert, A hierarchy of propositions: deciding which level to address in casework, Sci. Justice 38 (4) (1998) 231–239.

[6] Tom G. Schotman, Jaap van der Weerd, On the recovery of fibres by tape lifts, tape scanning, and manual isolation, Sci. Justice 55 (6) (2015) 415–421.

[7] Kelly J Sheridan, Ray Palmer, David A Chalton, Jariel N Bacar, Jack Beckett, Kieran Bellerby, Lucy Brown, Emily Donaghy, Alexander Finlayson, Cameron Graham, Beth Robertson, Lauren Taylor, Matteo D Gallidabino, A quantitative assessment of the extent and distribution of textile fibre transfer to persons involved in physical assault, Sci. Justice 63 (4) (2023) 509–516.

[8] T. Biermann, Fibre finder systems, in: Forensic Examination of Fibres, in: Forensic Science Series, Taylor & Francis, London, 1999, pp. 135–152.

[9] Shuttle PCP, Scientific High-throughput and Unified Toolkit for Trace analysis by forensic Laboratories in Europe.

[10] Mohamed Elgendy, Deep Learning for Computer Vision, vol. 6, Manning Publications, 2019.

[11] Caio Henrique Pinke Rodrigues, Milena Dantas da Cruz Sousa, Michele Avila dos Santos, Percio Almeida Fistarol Filho, Jesus Antonio Velho, Aline Thais Bruni, Gunshot entrance recognition by artificial intelligence using computer vision, Forensic Sci. Int. 377 (2025) 112616.

[12] Caio Henrique Pinke Rodrigues, Milena Dantas da Cruz Sousa, Michele Avila dos Santos, Percio Almeida Fistarol Filho, Jesus Antonio Velho, Vitor Barbanti Pereira Leite, Aline Thais Bruni, Forensic analysis of microtraces using image recognition through machine learning, Microchem. J. 207 (2024) 111780.

[13] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, in: F. Pereira, C.J. Burges, L. Bottou, K.Q. Weinberger (Eds.), in: Advances in Neural Information Processing Systems, vol. 25, Curran Associates, Inc., 2012.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[15] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, Bernt Schiele, The cityscapes dataset for semantic urban scene understanding, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3213–3223.

[16] Holger Caesar, Jasper Uijlings, Vittorio Ferrari, Coco-stuff: Thing and stuff classes in context, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 1209–1218.

[17] Yejin Kim, Tae Sup Yun, How to classify sand types: A deep learning approach, Eng. Geol. 288 (2021) 106142.

[18] Jiang Xiaojia, Yang Mengjing, Quan Yongzhi, He Ya, Hair microscopic image classification method based on convolutional neural network, in: 2019 IEEE International Conference on Power, Intelligent Computing and Systems, ICPICS, 2019, pp. 433–438.

[19] Evan Shelhamer, Jonathan Long, Trevor Darrell, et al., Fully convolutional networks for semantic segmentation., IEEE Trans. Pattern Anal. Mach. Intell. 39 (4) (2017) 640–651.

[20] T. Biermann, Collection of fibres on a 1:1 scale, in: Proceedings of the 6th European Fibre Group Meeting, Dundee, 1998, pp. 44–47.

[21] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C Berg, Li Fei-Fei, ImageNet Large Scale Visual Recognition Challenge, Int. J. Comput. Vis. (IJCV) 115 (3) (2015) 211–252.

[22] Simon Kornblith, Jonathon Shlens, Quoc V. Le, Do better imagenet models transfer better? in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2661–2671.

[23] Yuanyuan Zhou, Ji Zhang, Jiao Huang, Kaifei Deng, Jianhua Zhang, Zhiqiang Qin, Zhenyuan Wang, Xiaofeng Zhang, Ya Tuo, Liqin Chen, et al., Digital whole-slide image analysis for automated diatom test in forensic cases of drowning using a convolutional neural network algorithm, Forensic Sci. Int. 302 (2019) 109922.

[24] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al., Pytorch: An imperative style, high-performance deep learning library, Adv. Neural Inf. Process. Syst. 32 (2019).

[25] Saeed Shurrab, Rehab Duwairi, Self-supervised learning methods and applications in medical imaging analysis: A survey, PeerJ Comput. Sci. 8 (2022) e1045.

[26] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al., Bootstrap your own latent-a new approach to self-supervised learning, Adv. Neural Inf. Process. Syst. 33 (2020) 21271–21284.

[27] Colorado J Reed, Xiangyu Yue, Ani Nrusimha, Sayna Ebrahimi, Vivek Vijaykumar, Richard Mao, Bo Li, Shanghang Zhang, Devin Guillory, Sean Metzger, et al., Self-supervised pretraining improves self-supervised pretraining, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2022, pp. 2584–2594.

[28] Costas Balas, Charalampos Boras, Christos Chousos, Nikolaos Kaminakis, Nathanail Kortsalioudakis, Christos Rossos, Athanasios Tsapras, Christos Tsiaousis, A novel, multimodal, high throughput screening microscope platform for automating the analysis a of vast populations of forensics traces, in: 2023 IEEE International Workshop on Technologies for Defense and Security (TechDefense), IEEE, 2023, pp. 116–120.

[29] Peter Bankhead, Maurice B. Loughrey, José A. Fernández, Yvonne Dombrowski, Darragh G. McArt, Philip D. Dunne, Stephen McQuaid, Ronan T. Gray, Liam J. Murray, Helen G. Coleman, Jacqueline A. James, Manuel Salto-Tellez, Peter W. Hamilton, QuPath: Open source software for digital pathology image analysis, Sci. Rep. 7 (1) (2017).

[30] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, Ross Girshick, Momentum contrast for unsupervised visual representation learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 9729–9738.

[31] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, Alan L Yuille, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, IEEE Trans. Pattern Anal. Mach. Intell. 40 (4) (2017) 834–848.

[32] Ronneberger Olafand Fischer, Philippand Brox Thomas, U-Net: Convolutional Networks for Biomedical Image Segmentation, in: Navab Nassir, Joachim Hornegger, Wells William M., Frangi Alejandro F (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, Springer International Publishing, Cham, 2015, pp. 234–241.

[33] Qiang Wu, Fatima A. Merchant, Kenneth R. Castleman, Microscope Image Processing, Academic Press, 2008.

[34] Gregory W. Zack, William E. Rogers, Samuel A. Latt, Automatic measurement of sister chromatid exchange frequency., J. Histochem. Cytochem. 25 (7) (1977) 741–753.

[35] Emma A Levin, Ruth M Morgan, Lewis D Griffin, Vivienne J Jones, A comparison of thresholding methods for forensic reconstruction studies using fluorescent powder proxies for trace materials, J. Forensic Sci. 64 (2) (2019) 431–442.

[36] Paul L. Rosin, Unimodal thresholding, Pattern Recognit. 34 (11) (2001) 2083–2096.

[37] Ozan Ciga, Tony Xu, Anne Louise Martel, Self supervised contrastive learning for digital histopathology, Mach. Learn. Appl. 7 (2022) 100198.

[38] Michael Majurski, Peter Bajcsy, Exact tile-based segmentation inference for images larger than gpu memory, J. Res. Natl. Inst. Stand. Technol. 126 (2021) 1–16.

[39] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, Andrew Zisserman, The PASCAL visual object classes (VOC) challenge, Int. J. Comput. Vis. 88 (2010) 303–338.

[40] Juri Opitz, Sebastian Burst, Macro F1 and macro F1, 2021.

[41] Connor Shorten, Taghi M. Khoshgoftaar, A survey on image data augmentation for deep learning, J. Big Data 6 (1) (2019) 1–48.

[42] Ting Chen, Simon Kornblith, Mohammad Norouzi, Geoffrey Hinton, A simple framework for contrastive learning of visual representations, in: International Conference on Machine Learning, 2020, pp. 1597–1607.

[43] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, Lei Li, Dense contrastive learning for self-supervised visual pre-training, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 3024–3033.

[44] Chaitanya Patel, Shashank Sharma, Valerie J Pasquarella, Varun Gulshan, Evaluating self and semi-supervised methods for remote sensing segmentation tasks, 2021, arXiv preprint arXiv:2111.10079.

[45] Zhenda Xie, Yutong Lin, Zheng Zhang, Yue Cao, Stephen Lin, Han Hu, Propagate yourself: Exploring pixel-level consistency for unsupervised visual representation learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 16684–16693.